STUDY AND DEVELOPMENT OF SOME NOVEL FUZZY IMAGE SEGMENTATION TECHNIQUEs

A thesis submitted in partial fulfillment of the requirements for the degree of

MASTER OF TECHNOLOGY (Research)
In

ELECTRONICS AND COMMUNICATION ENGINEERING
By

KUMARI NIRULATA

Under the supervision
Of

DR. S. MEHER

Department Of Electronics and Communication Engineering
National Institute of Technology, Rourkela, India
August 2009

Certificate

*This is to certify that the thesis titled "**Study and Development of Some Novel Fuzzy Image Segmentation Techniques**", submitted to National Institute of Technology,Rourkela (INDIA) by **Kumari Nirulata**, roll no. 60609002 for the award of the degree of master of Technology in Electronics and Communication Engineering, is a bonafide record of the research work carried out by her under my supervision and guidance.*

The candidate has fulfilled all the requirements.

The thesis, which is based on candidate's own work, has not been submitted elsewhere for a degree/diploma.

In my opinion, the thesis is of standard required of a M.Tech (R) degree in engineering.

To the best of my knowledge, Mrs. Nirulata bears a good moral character and decent behavior.

6/8/09

Dr. Sukadev Meher
Asst. Professor
Department Of Electronics and Communication Engineering
National Institute of Technology,
Rourkela-769008(INDIA)

PREFACE

Digital Image Processing, developed during last two and half decades, has become a very important subject in electronics and computer engineering. Computer vision and robotic vision is one of the many areas it encompasses. Image object identification and segmentation are the two sub-areas of image restoration.

The goal of image segmentation is partition of an image into a set of disjoint regions with uniform and homogeneous attributes such as intensity, color, tone or texture etc. In many real situations, for images, issues such as limited spatial resolution, poor contrast, overlapping intensities, noise and intensity inhomogenities introduce fuzziness in the object boundaries in the image. Due to this the fuzzy set theory was proposed, which produced the idea of partial membership of belonging described by a membership function.

Fuzzy rule based segmentation and various fuzzy clustering based segmentation has been implemented and developed. The proposed FCM based segmentation methods are tested extensively by subjective and objective evaluation. Under low noise conditions, though many FCM based segmentation methods are very good in terms of objective evaluations, the resulting output images of almost all methods give nearly equal visual quality. Hence efforts are made here to develop efficient filters for suppression of a uniform random noise under moderate and high noise conditions. The developed algorithm has also been applied to biomedical image segmentation.

Therefore, the present research work may be treated as

(i) **developmental work**; and

(ii) **applied research work**.

I would be happy to see other researchers using the results reported in the thesis for developing better image filters. Moreover, I will be contended to find these filters implemented for practical applications in near future.

Kumari Nirulata

ACKNOWLEDGEMENT

I express my indebtedness and gratefulness to my teacher and supervisor Prof. Sukadev Meher for his continuous encouragement and guidance. I needed his support, guidance and encouragement throughout the research period. I am obliged to him for his moral support through all the stages during this doctoral research work. I am indebted to him for the valuable time he has spared for me during this work.

I am thankful to Prof. S. K. Patra, Head, Department of Electronics & Communication Engineering who provided all the official facilities to me. I am also thankful to other DSC members, Prof. G. Panda , Prof. K.K. Mahapatra and Prof. B. Majhi for their continuous support during the doctoral research work.

I would like to thank all my colleagues and friends N. Bhoi, R. Kulkarni, C.S. Rawat, Devi, Mamta, Satyasai and Sitanshu, for their company and cooperation during this period.

I take this opportunity to express my regards and obligation to my parents whose support and encouragement I can never forget in my life.

I would like to thank my husband Lalit for his patience and cooperation. I duly acknowledge the constant moral support he provided throughout.

Lastly, I am thankful to all those who have supported me directly or indirectly during the research work.

Kumari Nirulata

BIO-DATA OF THE CANDIDATE

Name of the candidate	:	Kumari Nirulata
Father's Name	:	Raghubansh Ku. Singh
Present Address	:	M.Tech(R) Scholar, Dept. of Electronics and Communication Engg. National Institute of Technology, Rourkela-769008
Permanent Address	:	Qr. No. B/7 N.I.T.Campus Rourkela- 769008

ACADEMIC QUALIFICATION :

(i) **B. E.** in Electronics and Instrumentation, Purushottam Institute of Engg. and Technology

BPUT, Rourkela, Orissa, INDIA

PUBLICATION:

(i) Published 02 papers in International Journals;

(ii) Communicated 01 papers to International Journals;

(iii) Published 03 papers in National and International Conferences.

CONTENTS

Abstract

Some fuzzy technique based segmentation methods are studied and implemented and some fuzzy c means clustering based segmentation algorithms are developed in this thesis to suppress high and low uniform random noise. The reason for not developing fuzzy rule based segmentation method is that they are application dependent

In many occasions, the images in real life are affected with noise. Fuzzy c means clustering based segmentation does not give good segmentation result under such condition. Various extension of the FCM method for segmentation are present in the literature. But most of them modify the objective function hence changing the basic FCM algorithm present in MATLAB toolboxes. Hence efforts have been made to develop FCM algorithm without modifying their objective function for better segmentation .

The fuzzy technique based segmentation methods that are studied and developed are summarized here.

(A) Fuzzy edge detection based segmentation: Two fuzzy edge detection methods are studied and implemented for segmentation: *(i) FIS based edge detection* and *(ii) Fast multilevel fuzzy edge detector (FMFED).*

(i): The **Fuzzy Inference system (FIS)** based edge detector consists of some fuzzy inference rules which are defined in such a way that the FIS system output ("edges") is high only for those pixels belonging to edges in the input image. A robustness to contrast and lightining variations were also taken into consideration while developing these rules.The output of the FIS based edge detector is then compared with the existing Sobel, LoG and Canny edge detector results. The algorithm is seen to be application dependent and time consuming.

(ii) Fast Multilevel Fuzzy Edge Detector: To realise the fast and accurate detection of edges, the FMFED algorithm is proposed. It first enhances the image contrast by means of a fast multilevel fuzzy enhancement algorithm using simple transformation function based on two image thresholds. Second, the edges are extracted from the enhanced image by using a two stage edge detector operator that identifies the edge candidates based on

local characteristics of the image and then determines the true edge pixels using edge detector operator based on extremum of the gradient values.

Finally the segmentation of the edge image is done by morphological operator by edge linking.

(B) FCM based segmentation: Two fuzzy clustering based segmentation methods are developed: *(i) Modified Spatial Fuzzy c-Means (MSFCM) (ii) Neighbourhood Attraction Fuzzy c-Means (NAFCM).* .

(i) Contrast-Limited Adaptive Histogram Equalization Fuzzy c-Means (CLAHEFCM): This proposed algorithm presents a color segmentation process for low contrast images or unevenly illuminated images. The algorithm presented in this paper first enhances the contrast of the image by using contrast limited adaptive histogram equalization. After the enhancement of the image this method divides the color space into a given number of clusters, the number of cluster are fixed initially. The image is converted from RGB color space to LAB color space before the clustering process. Clustering is done here by using Fuzzy c means algorithm. The image is segmented based on color of a region, that is, areas having same color are grouped together. The image segmentation is done by taking into consideration, to which cluster a given pixel belongs the most. The method has been applied on a number of color test images and it is observed to give good segmentation results

(ii) Modified Spatial Fuzzy c-means (MSFCM): The proposed algorithm divides the color space into a given number of clusters, the number of cluster are fixed initially. The image is converted from RGB color space to LAB color space before the clustering process. A robust segmentation technique based on extension to the traditional fuzzy c-means (FCM) clustering algorithm is proposed. The spatial information of each pixel in an image has been taken into consideration to get a noise free segmentation result. The image is segmented based on color of a region, that is, areas having same color are grouped together. The image segmentation is done by taking into consideration, to which cluster a given pixel belongs the most. The method has been applied to some color test images and its performance has been compared to FCM and FCM based methods to show its superiority over them. The proposed technique is observed to be an efficient and easy method for segmentation of noisy images.

(iv)**Neighbourhood Attraction Fuzzy c Means Algorithm:** A new algorithm based on the IFCM neighbourhood attraction is used without changing the distance function of the FCM and hence avoiding an extra neural network optimization step for the adjusting parameters of the distance function, it is called *Neighborhood Atrraction FCM (NAFCM)*. During clustering, each pixel attempts to attract its neighbouring pixels towards its own cluster. This neighbourhood attraction depends on two factors: the pixel intensities or feature attraction, and the spatial position of the neighbours or distance attraction, which also depends on neighbourhood structure. The NAFCM algorithm is tested on a synthetic image (chapter 6, figure 6.3-6.6) and a number of skin tumor images. It is observed to produce excellent clustering result under high noise condition when compared with the other FCM based clustering methods.

List of Abbreviations used

Abbreviations	
1. *FIS*	Fuzzy Inference System
2. *FID*	Fuzzification, Inference and de-fuzzification
3. *FCM*	Fuzzy c-means
4. *sFCM*	Spatial Fuzzy c means
5. *MSFCM*	Modified spatial fuzzy c means
6. *FMFED*	Fast Multilevel Fuzzy Edge Detection
7. *FMFE*	Fast Multilevel Fuzzy Enhancement
8. *ANFIS*	Adaptive Neuro Fuzzy Inference System
9. *FMMIS*	Fuzzy Min-Max Neural Network for Image Segmentation
10. *DH*	Result of applying h_{DH}
11. *DV*	Result of applying h_{DV}
12. *E*	Edge detected image
13. *FRIST*	Fuzzy Rules for Image Segmentation incorporating Texture features
14. *GFRIS*	Generic Fuzzy Rule based Image Segmentation
15. *PFCM*	Penalized Fuzzy c-means
16. *FMCM*	Fuzzy Membership c means
17. *SWFCM*	Spatially Weighted Fuzzy c-means
18. *k-NN*	k-nearest neighbour
19. *LoG*	Laplacian of Gaussian
20. *AHE*	Adaptive Histogram equalization

21. *CLAHE* Contrast-Limited Adaptive Histogram Equalization
22. *NC* Noisy Clustering
23. *PCM* Possibilistic c-means
24. *NAFCM* Neighbourhood Attraction Fuzzy c means
25. *IFCM* Improved Fuzzy c means
26. *HVS* Human Visual System

List of Symbols used

	Symbols	
1.	R	Entire image region
2.	R1,R2….Rn	N sub regions of R
3.	$P(Ri)$	Logical predicate defined over point in set Ri
4.	R'	Response of mask at any point in image
5.	w	Filtering mask
6.	z	Gray level of pixels
7.	T_E	Execution Time
8.	T	Non negative threshold
9.	$H_s(x, y)$	Impluse response of Gaussian function
10.	$F(x,y)$	Image in spatial domain
11.	$H(x,y)$	LoG function
12.	h_{DH}	Sobel operator for derivative in horizontal direction
13.	h_{DV}	Sobel operator for derivative in vertical direction
14.	h_{HP}	3x3, High pass filter
15.	h_{MF}	5x5, arithmetic mean filter
16.	H	Output of applying h_{HP}
17.	M_o	Mean value for object pixels
18.	M_b	Mean value for background pixels.
19.	S_o	Sum of objects pixels
20.	S_b	Sum of background pixels
21.	μ_{ij}	Fuzzy membership function where i and j are row and column value of image
22.	$E(\mu_{ij})$	Enhanced image using fuzzy enhancement operator
23.	s_{ij}	Edge sign
24.	l = 0,1..7	8 sub-windows

25.	$d^{l}_{i,j}$	Gradient value for pixel (i,j) in l sub-window
26.	d_{ij}	Final edge image
27.	c	Number of clusters
28.	m	the weighting exponents, 1 for ‘hard’ clustering, and increasing for fuzzier clustering
29.	$d^2(x_k, v_i)$	The distance measure between object xk and cluster center vi;
30.	n	Total number of pixels in image;
31.	u_{ik}	Fuzzy membership value of pixel k in cluster i;
32.	v_i	Cluster center for subset i in feature space
33.	U	Fuzzy c-partition matrix

CHAPTER1

Introduction

1

Preview

Digital image processing is the use of computer algorithms to perform image processing on digital images. As a subfield of digital signal processing, digital image processing has many advantages over analog image processing; it allows a much wider range of algorithms to be applied to input data, and can avoid problems such as the build-up of noise and signal distortion during processing. Image segmentation refers to the process of partitioning a digital image into multiple regions (set of pixels). The goal of segmentation is to simplify and/or change the representation of an image into something that is more meaningful and easier to analyse. Image segmentation is typically used to locate objects and boundaries (lines, curves, etc.) in an image.

In this thesis the various popular fuzzy techniques for image segmentation are studied. Various methods for better clustering and segmentation have been developed. The algorithms or methods developed are meant for online and real time applications like television, camera phone, etc.

1.1 Fundamentals of Digital Image Processing

Digital image processing is a subset of the electronic domain wherein the image is converted to an array of small integers, called *pixels* (derived from *picture element)*, representing a physical quantity such as scene radiance, stored in a digital memory, and processed by computer or other digital hardware. Digital image processing, either as enhancement for human observers or performing autonomous analysis, offers advantages in cost, speed, and flexibility, and with the rapidly falling price and rising performance of personal computers it has become the dominant method in use.

An image is denoted by two dimensional functions of the form $f(x,y)$. The value or amplitude of f at spatial coordinates (x,y) is a positive scalar quantity whose physical meaning is determined by the source of the image. In a digital image, (x,y), and the magnitude of f are all finite and discrete quantities.

It is a hard task to distinguish between the domains of image processing and any other related area such as computer vision. But the two areas are quite different in the kind of output we get from them. **Computer vision** is the science and technology of machines that see. As a scientific discipline, computer vision is concerned with the theory for building artificial systems that obtain information from images. The image data can take many forms, such as a video sequence, views from multiple cameras, or multi-dimensional data from a medical scanner. In computer vision, the input is a digital image and the output is some representation of its interesting features. Image processing is often used in computer vision as a pre-processing step. Image processing is defined as an area when both input and output are images.

As a technological discipline, computer vision seeks to apply the theories and models of computer vision to the construction of computer vision systems.

The organization of a computer vision system is highly application dependent. Some systems are stand-alone applications which solve a specific measurement or detection problem, while other constitute a sub-system of a larger design which, for example, also contains sub-systems for control of mechanical actuators, planning, information databases, man-machine interfaces, etc. The specific implementation of a computer vision system also depends on if its functionality is pre-specified or if some part of it can be learned or modified during operation. There are, however, typical functions which are found in many computer vision systems.

1. Image acquisition: A digital image is produced by one or several image sensor which, besides various types of light-sensitive cameras, includes range sensors, tomography devices, radar, ultra-sonic cameras, etc. Depending on the type of sensor, the resulting image data is an ordinary 2D image, a 3D volume, or an image sequence. The pixel values typically correspond to light intensity in one or several spectral bands (gray images or colour images), but can also be related to various physical measures, such as depth, absorption or reflectance of sonic or electromagnetic waves, or nuclear magnetic resonance.

2. Pre-processing: Before a computer vision method can be applied to image data in order to extract some specific piece of information, it is usually necessary to process the data in order to assure that it satisfies certain assumptions implied by the method. Examples are

(a) Re-sampling in order to assure that the image coordinate system is correct.

(b) Noise reduction in order to assure that sensor noise does not introduce false information.

(c) Contrast enhancement to assure that relevant information can be detected.

(d) Scale space representation to enhance image structures at locally appropriate scales.

3. Feature extraction: Image features at various levels of complexity are extracted from the image data. Typical examples of such features are

(a) Lines, edges and ridges.

(b) Localized interest points such as corners, blobs or points.

More complex features may be related to texture, shape or motion.

4. Detection/Segmentation: At some point in the processing a decision is made about which image points or regions of the image are relevant for further processing. Examples are

(a) Selection of a specific set of interest points

(b) Segmentation of one or multiple image regions which contain a specific object of interest.

5. High-level processing: At this step the input is typically a small set of data, for example a set of points or an image region which is assumed to contain a specific object. The remaining processing deals with, for example:

(a) Verification that the data satisfy model-based and application specific assumptions.

(b) Estimation of application specific parameters, such as object pose or object size.

(c) Classifying a detected object into different categories

Hence it can be said that *image segmentation* forms an integral part of computer vision systems and *is more an area of computer vision than image processing*.

1.2 Image Segmentation

1.2.1 Theory

Segmentation of an image entails the division or separation of the image into regions of similar attribute. The basic attribute for segmentation is image amplitude- luminance for a monochrome image and color components for a color image. Image edges and textures are also useful attributes for segmentation. The result of image segmentation is a set of regions that collectively cover the entire image, or a set of contours extracted from the image.

Segmentation does not involve classifying each segment. The segmentor only subdivides an image; it does not attempt to recognise the individual segments or their relationships to one another.

There is no theory of image segmentation. As a consequence, no single standard method of image segmentation has emerged. Rather, there are a collection of ad hoc methods that have received some degree of popularity. Because the methods are ad hoc, it would useful to have some means of assessing their performance. Haralick and Shapiro (1) have established the

following qualitative guidelines for “good” image segmentation:

(a) Regions of the image segmentation should be uniform and homogeneous with respect to some characteristic such as gray tone or texture.

(b) Region interiors should be simple and without many small holes

(c) Adjacent regions of segmentation should have significantly different values with respect to the characteristic on which they are uniform.

(d) Boundaries of each segment should be simple, not ragged, and must be spatially accurate.

1.2.2 Applications of segmentation

Some of the practical applications of image segmentation are:

1. Medical Imaging

- Locate tumors and other pathologies
- Measure tissue volumes
- Computer guided surgery
- Diagnosis
- Treatment planning
- Study of anatomical structures

2. Locate objects in satellite images (roads, forests, etc.)
3. Face recognition
4. Fingerprint recognition
5. Automatic traffic controlling systems
6. Machine vision

1.3 Literature survey of fuzzy techniques applied for segmentation

Fuzzy technique has been applied for various methods used for image segmentation. Fuzzy image segmentation is increasing in popularity because of rapid extension of fuzzy set theory, the development of various fuzzy set based mathematical modelling, synergistic combination of fuzzy, genetic algorithm and neural network[50],[51], and its successful and practical application in image processing, pattern recognition and computer vision system.

In this work fuzzy edge detector and fuzzy clustering based image segmentation are studied. Fuzzy based edge detection methods are extensively used for image segmentation. Efficient fuzzy technique based edge detection method which would yield good segmentation results on application of some edge tracking techniques and some times even without application of edge tracking methods have been discussed.

Tood law, Hidenori Itoh and Hirohisa seki [1] characterized the problem of detecting edges in images as a fuzzy reasoning problem. The edge detection problem is divided into three stages: filtering, detection, and tracing. It was finally concluded in the paper that the algorithm was able to assemble edge information in a meaningful way. Fuzzy reasoning based edge detection has also been popular for edge detection of images affected by noise [2- 4].

Olga Regina Pereira Bellon et al. [5] presented a methodology to perform edge detection in range images in order to provide a reliable and meaningful edge map, which helps to guide and improve range image segmentation by clustering technique. The obtained edge map leads to three important improvements: (1) the definition of the ideal number of regions to initialize the clustering algorithm; (2) the selection of suitable initial cluster centers; and (3) the successful identification of distinct regions with similar features.

Xiaohan Yu, J. Yla-Jaaski et al. [6] proposed a new method for texture segmentation based on edge detection. The new scheme is based on the idea that texture features change abruptly near boundaries between different textures, and the segmentation can be carried out by detecting the feature changes or so-called feature edges. In this algorithm, the image is first projected onto a hyperplane called the characteristic image, in which the value of each pixel is not a grey level but

a vector value of the local textural features. An edge detection algorithm is then extended to the vector space and applied to the hyperplane to detect the feature edges.

Liu Yi, Chen Xue-quan [7] presented an improved edge detection algorithm for remote sensing images, which is based on fuzzy logic theory and conventional Pal. King algorithm. The membership function was redesigned, the method of fuzzy enhancement was modified and an edge evaluation criteria was used to control the iterative procedure automatically. The presented algorithm was found to be superior to other edge detectors in edge detection of remote sensing images.

Jinbo Wu, Zhouping Yin, and Youlon Xiong [8] proposed a fast and accurate edge detection method for blurry images. The algorithm called fast multilevel fuzzy edge detection (FMFED) first enhances the image contrast by means of the fast multilevel fuzzy enhancement (FMFE) algorithm using the simple transformation function based on two image thresholds. Secondly, the

Edges are extracted from the enhanced image by a two-stage edge detection operator that identifies the edge candidates based on the local characteristics of the image.

Cristiano Jacques Miosso and Adolfo Bauchspiess [9] evaluated the performance of a fuzzy inference system in edge detection. It was concluded that despite the much superior computational effort when compared to the Sobel operator, the implemented FIS system presents greater robustness to contrast and lighting variations, besides avoiding obtaining double edges. Further tuning of the weights associated to the fuzzy inference rules is still necessary to reduce even more inclusion in the output image of pixels not belonging to edges.

Image thresholding is another method which is used for image segmentation. Fuzzy techniques are applied for this method.

Farrah wong HT, Ramachandran Nagaranjan et al. [10] presented an image segmentation method by using a threshold value determined by fuzzy logic. The fuzzy based segmentation reported in the paper is an automated threshold calculation. The threshold value calculated by utilizing the histogram of the image and the measure of fuzziness constitute the initial step in the proposed segmentation procedure. The threshold value is then used as an input for the split and merge method of segmentation. Wen-Bing, Jin-Wn Tian et al. [11] have presented a three level thresholding method for image segmentation based on probability partition, fuzzy partition and entropy theory. The procedure for finding the optimal combination of all the fuzzy parameters is implemented by a genetic algorithm with appropriate coding method so as to avoid useless chromosomes. M. Cheriet, J.N.Said et al.[12] presented a general recursive approach for image segmentation by extending Otsu's method. This approach segments the brightest homogeneous objects after the last recursion. There are many thresholding based image segmentation methods [13,14]. Most of these greyscale based segmentation methods often assume that the image has a uniform and stationary or quasistationary distribution of greyscale for various targets or background. So they are often not so effective for the images with complex structure because of the complex distribution of the greyscale of images. Some techniques [15] assume images to be mostly nonstationary with space variant distribution. The segmentation methods based on this model are dependent on local area. The performance of such local operator will degrade quickly as the noise increases.

The most important fuzzy based approach to image segmentation are: fuzzy clustering algorithms, fuzzy rule based approach and measure of fuzziness.

Lior Shamir[16] has described a human perception based approach to pixel color segmentation. Fuzzy sets are defined on the H, S and V components of the HSV color space and provide a fuzzy logic model that aims to follow the human intuition of color classification. The knowledge-driven model allows simple modification of the classification based on needs of a specific application, and the efficiency of the algorithm in terms of the computational complexity makes the proposed method suitable for applications where efficiency is a primary issue.

A. Borji and M. Hamidi [17] have proposed a new method for color image segmentation using fuzzy logic where they automatically produce a system for color classification and image segmentation with least number of rules and minimum error rate. A comprehensive learning particle swarm optimization technique is used to find optimal fuzzy rules and membership functions as it discourages premature convergence. Less computational load is needed when using this method compared to other methods like

ANFIS. Large train data set and its variety makes the proposed method invariant to illumination noise.

Estevez Pablo A., Flores Rodrigo J. et al. [18] proposed a method called FMMIS (fuzzy min-max neural network for image segmentation). The FMMIS method grows boxes from a set of seed pixels, to find the minimum bounded rectangle for each object present in the images. The proposed method is very fast and it may be applied to real-time image segmentation tasks.

G. Karmakar ,L. Dooley et al. [19] proposed a new algorithm called fuzzy rules for image segmentation incorporating texture features (FRIST), which includes two additional membership functions to those already defined in GFRIS(generic fuzzy rule based image segmentation). FRIST incorporates the fractal dimension and contrast features of a texture by considering image domain specific information. FRIST exhibits considerable improvement in the results obtained compared with the GFRIS approach for many different image types.

Tie Qi Chen and Yi Lu [20] developed a fuzzy clustering algorithm that iteratively generates color clusters using a uniquely defined fuzzy membership function and an objective function for clustering optimization. The region segmentation algorithm merges clusters in the image domain based on color similarity and spatial adjacency. Martin Tabakov [21] described a way of medical image segmentation using an appropriately defined fuzzy clustering method based on a fuzzy relation. The considered relation is defined in terms of Euclidean distance.

Ahmed Mohamed N., Yamany Sameh M. et al. [22] presented an algorithm for fuzzy segmentation of MRI data and estimation of intensity inhomogenities using fuzzy logic. The algorithm is formulated by modifying the objective function of the standard fuzzy c-means algorithm to compensate for such inhomogenities and allow the labelling of a pixel to be influenced by the labels in its immediate neighbourhood.

Y. Yang, Ch.Zheng and P. Lin [23] presented a novel penalized fuzzy c-means (PFCM) algorithm for image segmentation. The algorithm is formulated by incorporating the spatial neighbourhood information into the original FCM with a penalty term. The penalty term is inspired by the neighbourhood expectation maximization algorithm and is modified in order to satisfy the criterion of the FCM algorithm. The algorithm is found to be more robust to noise than standard FCM.

Shan Shen,William Sandham et al. [24] presented an extension to the original FCM. The algorithm is based on neighbourhood attraction, which is dependent on the relative location and features of the neighbouring pixels. The degree of attraction is optimized by a neural-network model.

Jiayin Kang, Lequan Min et al. [25] presented a novel method for image segmentation by incorporating spatial neighbourhood information into the standard FCM. An adaptive weighted averaging filter is given to indicate the spatial influence of the center pixel.

Li Ma and R. C. Staunton [26] proposed a novel FCM algorithm to be used when active or structured lights are projected onto a scene. The recursive FCM algorithm is modified to include biased illumination field estimation. New clustering center and fuzzy clustering functions resulted based on the intensity and the average intensity of a pixel neighbourhood based object function. A dilation operator was used in the end on the initial segmented image for further refinement. The proposed method is found to be effective for segmenting images illuminated by patterns containing underlying biased intensity fields.

Yannis A. Tolias and Stavros M. Panas [27] presented the adaptive fuzzy clustering/segmentation (AFCS). In AFCS, the nonstationary nature of the images is taken into account by modifying the prototype vectors as function of sample location in the image. A multiresolution model is utilized for estimating the spatially varying prototype vectors for different window sizes. The segmentation of different resolutions is combined using a data fusion process in order to compute the final fuzzy partition matrix.The results provide segmentation having lower entropy.

N. A. Mohamed, M.N. Ahmed et al.[28] described the application of fuzzy set theory in medical imaging. A fully automatic technique to obtain clusters is proposed. A modified fuzzy c-means classification algorithm is used to provide a fuzzy partition. The method is inspired by Markov random Field (MRF) and is found to be less sensitive to noise as it filters the image while clustering it.

S R Kannan [29] presented a new method called fuzzy membership c-means(FMCM) for segmentation of Magnetic Resonance Images(MRI). This work develops a specific method to construct the initial membership matrix to clusters in order to improve the strength of the clusters.

Y. Yong, Z. Chongxun et al. [30] presented a spatially weighted fuzzy c-means (SWFM) clustering algorithm for image thresholding. Spatial neighbourhood information is taken into account in this algorithm. Two improved implementations of the k-nearest neighbour(k-NN) algorithm re introduced for calculating the weight in the SWFCM to improve thresholding. To speed up FCM algorithm the iteration is carried out on histogram of the image instead of all pixels of the image.

1.4 Problem Statement

In general, the classification of an image's pixel belonging to one of the "objects" (i.e., classes) composing the image is based on some common feature(s), or resemblance to some pattern. In order to determine which are the features that can lead to a successful classification, some apriori knowledge or/and assumptions about the image are equally required.

Classical, so-called "crisp" image segmentation techniques, while effective for images containing well-defined structures such as edges, do not perform well in the presence of ill-defined data. In such circumstances, the processing of images that posses ambiguity is better performed using fuzzy segmentation techniques, which are more adept at dealing with imprecise data. Fuzzy techniques may be broadly classified into five main categories:

1. Fuzzy clustering based image segmentation
2. Fuzzy rule based image segmentation
3. Fuzzy geometry based image segmentation
4. Fuzzy thresholding based image segmentation
5. Fuzzy integral based segmentation techniques (Tizhoosh,1998).

Of all these methods mentioned, the most widely used are the fuzzy rule based and fuzzy clustering based segmentation. The problem with fuzzy rule based image segmentation techniques is that they are application dependent with the structure of the membership functions being predefined and in certain cases, the corresponding parameters being manually determined. Karmakar et al. [76] presented a contemporary review of fuzzy rule based image segmentation techniques, and confirmed that despite being used in a wide range of applications, both the structure of membership functions and derivation of their relevant parameters were still very much application domain and image dependent. Fuzzy c-means is an unsupervised technique that has been successfully applied to feature analysis, clustering, and classifier designs in fields such as astronomy, geology, medical imaging, target recognition, and image segmentation [21]-[28],[61],[62],[74],[75]. An image can be represented in various feature spaces, and the FCM algorithm classifies the image by grouping similar data points in the feature space into clusters. This clustering is achieved by iteratively minimizing a cost function that is dependent on the distance of the pixels to the cluster centers in the feature domain.

Unfortunately, the greatest shortcoming of FCM is its over-sensitivity to noise, which is also a flaw of many other intensity based segmentation methods. In recent years, many modification of the FCM algorithm have been reported to overcome the effect of noise.

Most of these methods inevitably introduce computation issues. In almost all methods proposed recently, the objective function of the FCM is changed. As most equations are modified along with the modification of the objective function, these methods lose continuity from FCM, which is well-realized with many types of software, such as MATLAB.

1.5 Image Metrics

The quality of an image is examined by objective evaluation as well as subjective evaluation. The subjective evaluation is the most widely used type of evaluation method, in which the segmentation results are judged by a human evaluator. The disadvantage of such methods is that visual or qualitative evaluation is inherently subjective. Subjective evaluation scores may vary significantly from one human evaluator to another, because each evaluator has their own distinct standards for assessing the quality of a segmented image. The image metrics for fuzzy clustering based segmentation are discussed here. In fuzzy clustering based method good clustering of the image amounts to good segmentation. Hence in order to obtain a quantitative comparison, two types of cluster validity functions, fuzzy partition and feature structure, are often used to evaluate the performance of clustering in different clustering methods. The representative functions for the fuzzy partition are partition coefficient V_{pc} [31] partition entropy V_{pe} [32]. They are defined as follows:

$$V_{pc} = \frac{\sum_{k=1}^{n} \sum_{i=1}^{c} u_{ik}}{n} \tag{1.1}$$

and

$$V_{pe} = \frac{-\sum_{k=1}^{n} \sum_{i=1}^{c} [u_{ik} \log u_{ik}]}{n} \tag{1.2}$$

The value of V_{pc} is in the range [1/c,1]. An index close to 1 indicates good cluster separation, while a low index value indicates fuzzier clustering. An index of V_{pc} =1/c indicates that there is no clustering tendency. The value of V_{pe} is in the range [0,log c]. In contrast to V_{pc}, a low value of V_{pe} indicates good cluster separation. The idea of these validity functions is that the partition with less fuzziness means better performance. As a result, best clustering is achieved when the value V_{pc} is maximal or V_{pe} is minimal.

The third image metric used for comparision of different algorithms present and proposed is the percentage of misclassified pixels present in a class(cluster).To find the number of misclassified pixels in each clusters first we find the number of pixels in each clusters when noise is not added to the image. After that, we add noise to the image and calculate the number of pixels which are misclassified i.e the number of pixels that have increased in a cluster after adding noise or the missing pixels in a cluster after adding noise. Finally the percentage of misclassified pixels is calculated using the formula :

$$\frac{Number\,of\ misclassified\ pixels\ in\,a\,cluster}{Orignal\,number\,of\ pixels\ in\ the\,cluster} * 100 \tag{1.3}$$

Another image metric used for comparison of different methods is the execution time. Execution time is defined as the time taken for the simulation of an algorithm. The less time an algorithm takes for execution the more efficient it is considered. The processesor used is a Pentium IV core 2 duo processor, 2.4Ghz (clock), 2GB (RAM), Windows vista 64 bit operating system.

1.6 Conclusion

In this introductory chapter, the fundamentals of digital image processing, theory and application of image segmentation, the existing image segmentation techniques and their merits and demerits

and various image metrics are studied. The advantages and disadvantages of fuzzy rule based segmentation and fuzzy clustering based segmentation have been discussed.

Hence, it is decided to study and develop various fuzzy rule based segmentation method and fuzzy clustering based segmentation algorithms.

CHAPTER 2

Basic techniques of image segmentation

2

Preview

Image segmentation algorithms are generally based on one of the two basic properties of intensity values: discontinuity and similarity. In the first category, the approach is to partition an image based on abrupt changes in intensity, such as **edges** in an image. Segmentation based on discontinuity method is discussed in next chapter. The principal approaches in the second category are based on partitioning an image into regions that are similar according to a set of pre-defined criteria. Thresholding, region growing, and region splitting and merging are examples of methods in this category. Segmentation based on similarity property of intensity values that is region based segmentation methods are described here.

2.1 Region-Based Segmentation

2.1.1 Basic formulation:

Let R represent the entire image region. Segmentation may be viewed as a process that partitions R into n subregions, $R_1, R_2, ..., R_n$ such that

(a) $\bigcup_{i=1}^{n} R_i = R$

(b) R_i is a connected region, i= 1,2,….,n.

(c) $R_i \bigcap R_j = \varnothing$ for all i and j, $i \neq j$

(d) $P(R_i)$ = TRUE for all i = 1,2,….,n.

(e) $P(R_i \bigcup R_j)$ = FALSE for $i \neq j$

Here, $P(R_i)$ is a logical predicate defined over the points in set R_i and $\varnothing$ is the null set. Condition (a) indicates that the segmentation must be complete; that is, every pixel must be in a region. Condition (b) requires that points in a region must be connected in some predefined sense. Condition (c) indicates that the regions must be disjoint. Condition (d) deals with the properties that must be satisfied by the pixels in a segmented region- for example $P(R_i)$ = TRUE if all pixels in R_i have the same gray level. Finally, condition (e) indicates that regions R_i and Rj are different in the sense of the predicate P [33].

2.1.2 Region growing

Region growing is a procedure that group's pixels or subregions into larger regions based on predefined criteria [34]. The basic approach is to start with a set of "seed" points and from these grow regions by appending to each seed those neighboring pixels that that properties similar to the seed (such as specific ranges of gray level or color).

This approach has specific **advantages** over boundary based (pixel differences) methods:

1. It is guaranteed (by definition) to produce coherent regions. Linking edges, gaps produced by missing edge pixels, etc. are not an issue

2. It works from the inside out, instead of the outside in. The question which object a pixel belongs to, is immediate, not the result of point-in-contour tests.

However, it also has **drawbacks**:

1. Decisions about region membership are often more difficult than applying edge detectors.

2. It can't find objects that span multiple disconnected regions. (Whereas edge-based method can be designed to handle "gaps" produced by occlusion—the Hough transform is one example

The objectives of region-based approaches can be summarized as follows:

(a) Produce regions that are as large as possible (i.e., produce as few regions as possible).

(b) Produce coherent regions, but allow some flexibility for variation within the region.

2.1.2.1 How to choose the seed(s) for region growing in practice?

1. It depends on the nature of the problem.
2. If target need to be detected using infrared images for example, choose the brightest pixels
3. Without a-priori knowledge, compute the histogram and choose the gray-level values corresponding to the strongest peaks.

2.1.2.2 How to choose the similarity criteria (predicates)?

The homogeneity predicate can be based on any characteristic of the regions in the image such as

* Average intensity
* Variance
* Color
* Texture
* Motion
* Shape
* Size

Selecting a set of one or more starting points often can be based on the nature of the problem. When a priori information is not available, the procedure is to compute at every pixel the same set of properties that ultimately will be used to assign pixels to the regions during the growing process. If the result of these computations shows clusters of values, the pixels whose properties place them near the centroid of these clusters can be used as seeds.

The selection of similarity criteria depends not only on the problem under consideration, but also on the type of image data available. For example, the analysis of land-use satellite imagery depends heavily on the use of color. This problem would be significantly more difficult, or even impossible to handle without the inherent information available in color images. When the images are monochrome, region analysis must be carried out with a set of descriptors based on gray levels and spatial properties (such as moments and texture).

Descriptors alone may yield misleading results if connectivity or adjacency information is not used in the region-growing process.

Region growing should stop when no more pixels satisfy the criteria for inclusion in that region. Criteria such as gray level, texture, and colour, are local in nature and do not take into account the history of region growth. Hence the power of region growing algorithms are increased by utilizing the concept of size, likeness between a candidate pixel and the pixels grown so far (such as a comparison of the gray level of a candidate and the average gray level of the grown region), and the region being grown.

2.1.3 Region split and merge

Split and merge image segmentation techniques are based on a quad tree data representation whereby a square image is broken (split) into four quadrants if the original image segment is nonuniform in attribute. If four neighboring squares are found to be uniform, they are replaced (merge) by a single square composed of the four adjacent squares.

Subdivide an image initially into a set of arbitrary, disjoint regions and then merge and/or split the regions in an attempt to satisfy the necessary conditions

Let R represent entire image region and select a predicate P

(1) Split into four disjoint quadrants any region R_i for which $P(R_i)$ = FALSE

(2) Merge any adjacent regions R_j and R_k for which $P(R_j \cup R_k)$ = TRUE

(3) Stop when no further merging or splitting is possible

Several variations of this theme are possible

2.1.3.1 Quadtrees for region extraction

Important data structures which is used in split and merge algorithms is the quadtree. Figure 2.1 shows a quadtree and its relation to the image. Note that in graphics the quadtree is used in a region splitting algorithm (Warnock's Algorithm) which breaks a graphical image down recursively from the root node, which represents the whole image, to the leaf nodes where each leaf node represent a coherent region, which can be rendered without further hidden line elimination calculations[14]. The same use is made of quadtrees for vision. Quadtrees impose one type of regular decomposition onto an image. To complete the segmentation process this must be followed by a merging phase. Thus the problem of finding adjacent neighbours to a given node has been studied in figure 2.2. The problem is one of tree search and efficient algorithms have been published.

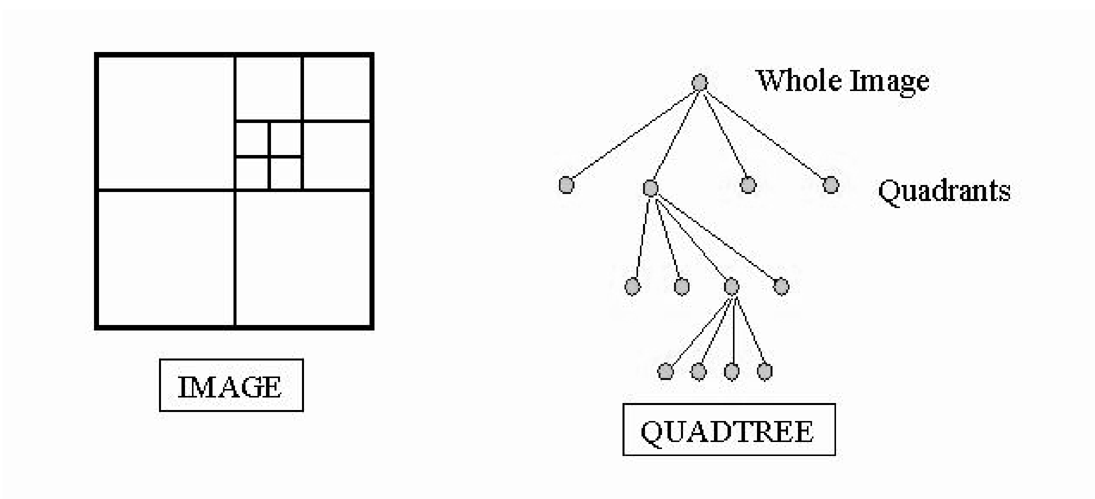


Figure 2.1 Quadtree decomposition

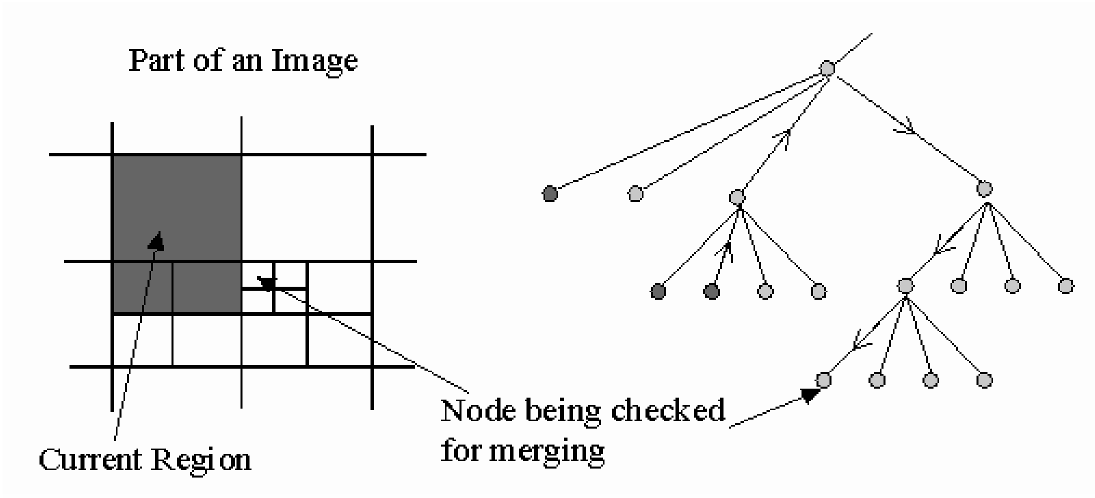


Figure 2.2 Splitting and merging with quadtrees

2.2 Segmentation technique based on discontinuity property of pixels.

2.2.1 Detection of Discontinuities

In this category, the approach is to partition an image based on abrupt changes in intensity, such as **edges** in an image. Three basic types of gray-level discontinuities that are mostly detected in a digital image are: points, lines and edges. The most common way to look for discontinuities is to run a mask through the image. For the 3x3 mask shown in fig. 3.1 , this procedure involves computing the sum of products of the coefficient with the gray level contained in the region encompassed by the mask. That is, the response of the mask at any point in the image is given by

$$R' = w_1 z_1 + w_2 z_2 + \ldots + w_9 z_9 = \sum_{i=1}^{9} w_i z_i \tag{2.1}$$

-1	-1	-1
-1	8	-1
-1	-1	-1

Figure 2.3 Point detection mask

where z_i is the gray level of the pixel associated with mask coefficient w_i. As usual, the response of the mask is defined with respect to its center location.

2.2.1.1 Point detection

Using the mask shown in Fig. 2.3, we say that a point has been detected at the location on which the mask is centered if

$$| R^{'} | \geq T \tag{2.2}$$

where T is a nonnegative threshold and R’ is given by (2.1).

2.2.1.2 Line detection

Consider the masks in Fig. 2.4. If the first mask were moved around an image, it would respond more strongly to lines (one pixel thick) oriented horizontally. With a constant background, the maximum response would result when the line passed through the middle row of the mask. Similarly, the second mask in Fig. 2.4 responds to lines oriented at $+45^{0}$; the third mask to vertical lines; and the fourth mask to lines oriented at -45^{0} direction.

Let R1’, R2’, R3’, and R4’ denote the responses of the masks in Fig. 2.4, from left to right, where R’s are given by equation 2.1. Let the four masks be run through an image individually. If, at a certain point in the image, |R*i'*|>|R*j'*|, for all $j \neq i$, that point is said to be more likely associated with a line in the direction of the mask *i*.

−1	−1	−1
2	2	2
−1	−1	−1

Horizontal

−1	−1	2
−1	2	−1
2	−1	−1

+ 45°

−1	2	−1
−1	2	−1
−1	2	−1

Vertical

2	−1	−1
−1	2	−1
−1	−1	2

−45°

Figure 2.4. Line detector masks

2.2.1.3 Edge detection

Edge detection is an important step for image segmentation. The goal of edge detection process in a digital image is to determine the frontiers of all represented objects based on automatic processing of the color or gray level information in each present pixel.

To extract the edges from the images, derivative edge detection operators or gradient operator, such as Sobel operator, Prewitt operator, Roberts operator, and Laplacian operators are commonly used. A 3x3 mask is used for edge detection using the mentioned operators. The various masks and the result of applying them on the image are shown in fig. 2.4 and fig. 3.5 respectively.

The reasons that Prewitt and Sobel edge detectors visually appear to better delineate object edges than the Roberts edge detector is attributable to their larger size, which provides averaging of small luminance fluctuations. The Sobel edge detector uses a weight of 2 in the center coefficient. A weight of 2 is used to achieve some smoothing by giving more importance to the center point. The Prewitt masks are simpler to implement than the Sobel masks, but the latter have slightly superior noise-suppression characteristics, an important issue when dealing with derivatives. Note that the coefficients in all masks shown in Fig. 2.5 sum to 0, indicating that they give a response of 0 in areas of constant gray levels, as expected of a derivative operator.

−1	0
0	1

0	−1
1	0

Roberts

−1	0	1
−1	0	1
−1	0	1

−1	−1	−1
0	0	0
1	1	1

Prewitt

−1	−2	−1
0	0	0
1	2	1

−1	0	1
−2	0	2
−1	0	1

Sobel

Figure 2.5. Line detection masks

(A) Laplacian of Gaussian edge detector

Marr and Hildreth [35] have proposed the Laplacian of Gaussian (LoG) edge detection operator operator in which Gaussian-shaped smoothing is performed prior to application of the Laplacian. The continuous domain LoG gradient is

$$G(x,y) = -\nabla^2 \left\{ F(x,y) \otimes H_s(x,y) \right\} \tag{2.3}$$

where

$$H_s(x,y)=g(x,s)g(y,s) \tag{2.4}$$

is the impulse response of the Gaussian smoothing function as defined by

$$g(x,s)=\left[2\pi s^2\right]^{-1/2}\exp\left\{-1/2(x/s)^2\right\} \tag{2.5}$$

where s is standard deviation

As a result of the linearity of the second derivative operation and of the linearity of convolution, it is possible to express the LoG response as

$$G(x,y)=F(x,y)\otimes H(x,y) \tag{2.6}$$

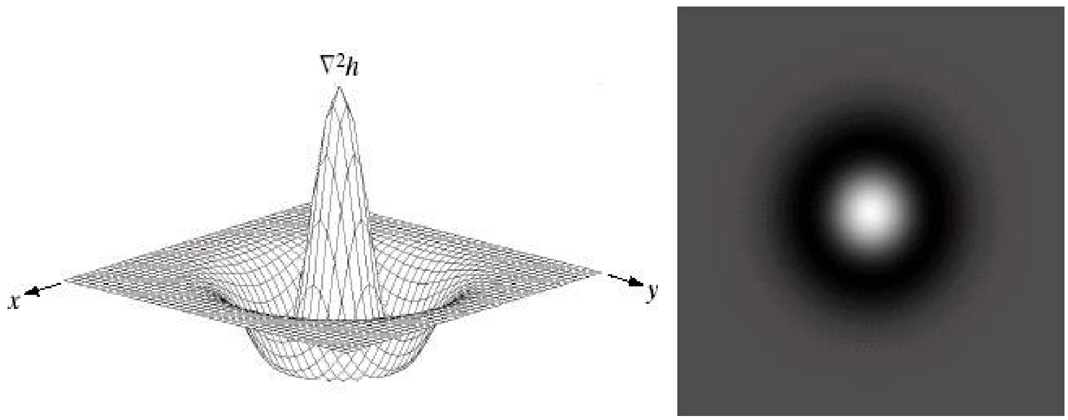
where

$$H(x,y)=-\nabla^2\left\{g(x,s)g(y,s)\right\} \tag{2.7}$$

Upon differentiation one obtains

$$H(x,y)=\frac{1}{\pi s^4}g(x,s)\left[1-\frac{x^2+y^2}{2s^2}\right]\exp\left\{-\frac{x^2+y^2}{2s^2}\right\} \tag{2.8}$$

This function is commonly referred to as the *Laplacian of a Gaussian* (LoG) because Eq.2.8 is in the form of a Gaussian function. A 5x5 mask that approximates $H(x,y)$ is shown in Fig.2.6(c). This approximation is not unique. Its purpose is to capture the essential shape of $H(x,y)$; that is, a positive central term, surrounded by an adjacent negative region that increases in value as a function of distance from the origin, and a zero outer region. The coefficients must also sum to zero, so that the response of the mask is zero in areas of constant gray level. Due to its shape, the Laplacian of Gaussian is called the *Mexican hat function*.



(a) (b)

0	0	−1	0	0
0	−1	−2	−1	0
−1	−2	16	−2	−1
0	−1	−2	−1	0
0	0	−1	0	0

(c)

Figure 2.6 Laplacian of a Gaussian (LoG). (a) 3-D plot. (b) Image (black is negative, gray is the zero plane, and white is positive). (c) 5x5 mask approximation to the shape of (a)

(B) Canny edge detector

The Canny edge detection operator was developed by John F. Canny [57] in 1986 and uses a multi-stage algorithm to detect a wide range of edges in images. The method can be summarized as follows:

1. The image is smoothed using a Gaussian filter with a specified standard deviation, s, to reduce noise.

2. The local gradient, $g(x,y)=[G_x^2+G_y^2]^{1/2}$, and edge direction, $\alpha(x,y)=\tan^{-1}(G_x/G_y)$, are computed at each point. Any of the first three techniques Prewitt, Sobel or LoG edge

detector can be used to compute G_x and G_y. An edge point is defined to be a point whose strength is locally maximum in the direction of the gradient.

3. The edge points determined in (2) give rise to ridges in the gradient magnitude image. The algorithm then tracks along the top of these ridges and sets to zero all pixels that are not actually on the ridge top so as to give a thin line in the output, a process known as *nonmaximal suppression.* The ridge pixels are then thresholded using two thresholds, T1 and T2, with T1<T2. Ridge pixels with values greater than T2 are said to be “strong” edge pixels. Ridge pixels with values between T1 and T2 are said to be “weak” edge pixels.

4. Finally, the algorithm performs edge linking by incorporating the weak pixels that are 8-connected to the strong pixels.

The gradient-based edge detection method has been widely applied in practice and a reasonable edge map is obtained for most images. Nevertheless, they suffer from some practical limitations.

First, they need a smoothing operation to alleviate the effect of high spatial frequency in estimating the gradient. Usually this smoothing is applied to all pixels in the image including the edge regions, and so the edge is distorted and missed in some cases, in particular at junctions or corners. Secondly, the gradient magnitude alone is insufficient to determine meaningful edges because of the ambiguity caused by the underlying pixel pattern, especially in complex natural scenes. Thirdly, the gradient-based edge detection methods increase the computational complexity because calculations, such as square root and arctangent, to produce the gradient vector are required. Finally, for edge thresholding conventional gradient methods use one or two global edge thresholds for an input image. For example, the hysteresis thresholding proposed by Canny in many practical applications require not only the trial and error adjustment of two thresholds to produce a satisfactory edge result for each different input image, but also the validity of the pre-adjusted thresholds.

The simulation results and conclusion of this chapter are in chapter 6.

CHAPTER 3

Study and Implementation of Segmentation based on Fuzzy Edge Detection

3

Preview

The goal of edge detection in image processing is to determine the frontiers of all represented objects, based on automatic processing of color or gray level information contained in each pixel. This procedure has many applications in image processing, computer vision and biological and robotic vision [46], [47], and [48].

Edge detection of real world images is a challenging task as there are a number of objects and huge variations between them which makes it difficult to approximate all objects using a general frame. Segmentation based on edge detection mostly consists of two steps:

1. Edge detection
2. Edge linking

Most real world images posses a certain amount of ambiguity and hence their segmentation produces fuzzy regions. For such images, fuzzy image segmentation techniques are more adept for processing their uncertainties. The importance of the fuzzy sets for analyzing complex natural

systems has been determined in several application domains. Digital images, which are mappings of natural scenes, are always accompanied by some degree of uncertainty (fuzziness) mainly due to:

i) Imprecision of gray values of the pixels;

ii) Ambiguity resulting from the image acquisition and mapping mechanism;

iii) Vague information in the region boundaries.

This fact justifies the development of algorithms based on fuzzy sets for several tasks of image analysis.

Recent techniques have characterized edge detection as a fuzzy reasoning problem [37], [38], [40], [41], [42]. These techniques have presented good and, therefore, promising results in the areas of image processing and computational vision. Fuzzy techniques allow a new perspective to model uncertainties due to the uncertainty of gray-values present in the images. Thus, instead of assigning gray-values to the pixels in the image, fuzzy membership may be used to the gray-values in the image.

Fuzzy approaches for image segmentation may be classified as approaches based on fuzzy rules; fuzzy classification algorithms; fuzziness measurements and image information and fuzzy geometry [39]. The approach based on rules treats image characteristics as linguistic variables and, therefore, uses IF-THEN fuzzy rules to segment images in different regions [36], [40], [41]. Fuzzy classification is the oldest approach for image segmentation. Algorithms such as the **c-means** fuzzy and possibilistic **c-means** may be used to build classes (segments) [40], [41], [27]. Fuzziness measurements (fuzzy entropy) and image information (fuzzy divergence) may also be used to segment images [38], [44].

3.1 A Fuzzy Inference System for Edge Detection based Segmentation

A nonlinear image filtering technique is developed here which is based on **fuzzy inference systems (FIS)** [45]. During input image processing, three kinds of linear filters are applied to it:

1. Sobel operators, used to estimate its derivative in horizontal and vertical directions (h_{DH} and h_{DV} filters)
2. A low-pass filter and
3. A high-pass filter.

Here the gray level associate to pixel (i,j) in the output image *E* depends not only on the pixel (i,j) in each pre-processed image but also on some neighbor pixels, as depicted in Figure 3.1. Besides, each image *DH* and *DV* that results from applying Sobel operators is passed to the FIS system, and not only the image composition $D = \sqrt{DH^2 + DV^2}$.

The purpose of proposed fuzzy system is to determine if pixel (i,j) evaluated is or is not present in one of the edges of the image, given the information explicit in the input filtered images.

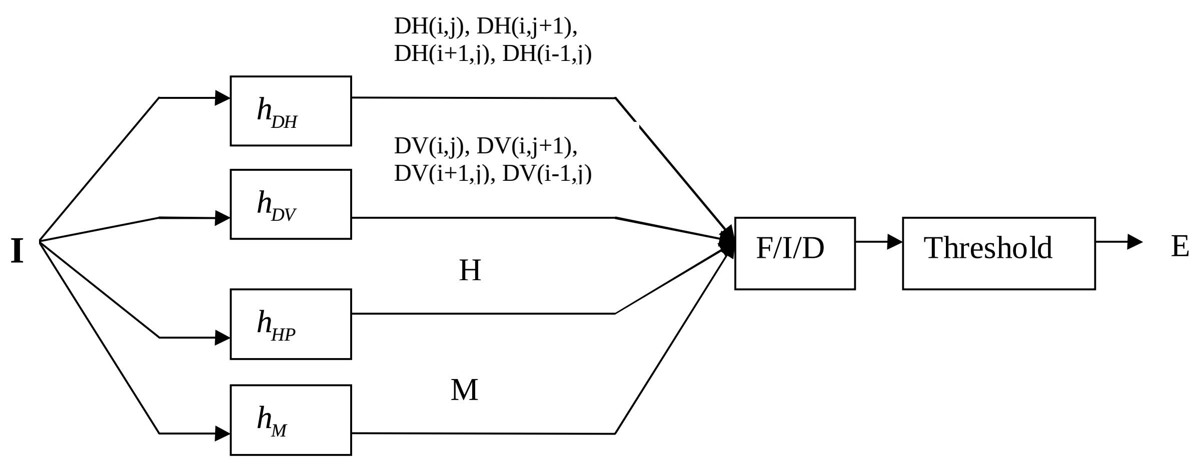


Figure 3.1. FIS applied to edge detection in image I. h_{DH} and h_{DV} are Sobel operators to estimate 1st derivative of I in horizontal and vertical directions. h_{HP} & h_M are masks of a high pass and low pass filters. F,I,D refer to fuzzification, inference and deffuzification stages.

3.1.1 Implementation of the FIS system

During input image pre-processing step, four linear filters were employed. Sobel operator h_{DH} and h_{DV} are masks of size 3x3 and are given by

h_{DH} =

−1	0	1
−2	0	2
−1	0	1

h_{DV} =

−1	−2	−1
0	0	0
1	2	1

The high pass filter mask is given by:

$$h_{HP} = \begin{bmatrix} -1/16 & -1/8 & -1/16 \\ -1/8 & 3/4 & -1/8 \\ -1/16 & -1/8 & -1/16 \end{bmatrix}$$

The low pass filter mask is selected in such a way that the gray level in each pixel of the output image is the arithmetic mean of the gray levels in a 5x5 neighbourhood of the same pixel in the input image.

The mask for low pass filter is given as

$$h_{MF} = \frac{1}{25}.\begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

Given the masks associated with each filter, the filtered images may be computed through a bi-dimensional convolution operation.

$$DH = h_{DH} * I$$

$$DV = h_{DV} * I$$

$$HP = h_{HP} * I$$

$$M = h_{MF} * I$$

3.1.2 Fuzzy sets and fuzzy membership functions

The system implementation was carried out considering that the input image and the output image obtained after *defuzzification* are both 8-bit quantized; this way, their gray levels are always between 0 and 255. These values define the working interval of the output variable and the input variable M (the other input variables are not guaranteed to be less than 255). Besides, three fuzzy sets were created to represent each variable's intensities; these sets were associated to the linguistic variables "low", "medium" and "high".

The Gaussian membership function is adopted for the fuzzy sets ("low , medium and high") associated with input M and the output. The mean value for the Gaussian membership function is taken as 0, 127.5 and 255 as shown in figure 3.2(a). For the fuzzy set associated with inputs DV,HP and output, Gaussian functions were also adopted for linguistic variables "low" and "medium". The membership function for linguistic variable "high" is chosen to be a sigmoid function, since in this case we can not guarantee that the input values will be restricted to the interval [0,255].

3.1.3 Fuzzy logical operations and defuzzification method definitions

The functions adopted to implement the "and" and "or" operations were the minimum and maximum functions, respectively. The Mamdani method was chosen as the defuzzification procedure, which means that the fuzzy sets obtained by applying each inference rule to the input data were joined through the add function; the output of the system was then computed as the centroid of the resulting membership function [52, pages 2-20 to 2-23].

3.1.4 Inference rules

The fuzzy inference rules were defined in such a way that the FIS system output ("Edges") is high only for those pixels belonging to edges in the input image.
The first three rules were defined to represent the general notion that in pixels belonging to an edge there is a high variation of gray level in the vertical or horizontal direction:

1. (DH low) AND (DV low) → ("Edges" low).
2. (DH medium) AND (DV medium) → ("Edges" high).
3. (DH high) OR (DV high) → ("Edges" high).

To guarantee that edges in regions of relatively low contrast can be detected, the two following rules were established to favour medium variations of the gray level in a specific direction in regions of *low frequency* of the input image (HP "low"):

4. (DH medium) AND (HP low) → (" Edges" high).
5. (DV medium) AND (HP low) → (" Edges" high).

To avoid including in the output image, pixels belonging to regions of the input where the mean gray level is lower, the following two rules were established. These regions are proportionally more affected by noise, supposed it is uniformly distributed over the whole image. The goal here
is to design a system which makes it easier to include edges in low contrast regions, but which does not favour false edges by effect of noise.

6. (DV medium) AND (M low) $\rightarrow$ ("Edges" low).
7. (DH medium) AND (M low) $\rightarrow$ ("Edges" low).

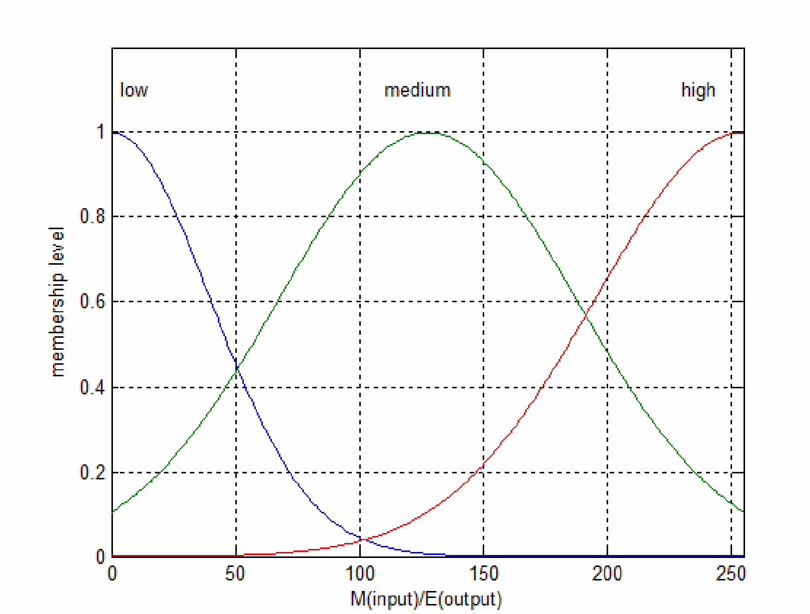
To avoid forming double edges in the output image that tend to appear due to shadows in the natural images, following four rules were developed. Considering that high variations in gray level in horizontal direction correspond to vertical edges, it is concluded that high values of DH(i,j) and DH $(i,j\pm1)$ do not imply edge pixels in (i,j) and $(i,j\pm1)$ simultaneously. High values of DV(i,j) and DV$(i\pm1,j)$ do not correspond to edge pixels in (i,j) and $(i\pm1,j)$.

8. (DV high) AND (DV (i + 1, j) high) $\rightarrow$ ("Edges" medium).
9. (DH high) AND (DH (i, j + 1) high) $\rightarrow$ ("Edges" medium).
10. (DV medium) AND (DV (i+1,j) high) $\rightarrow$ ("Edges" low).
11. (DH medium) AND (DH (i, j+1) high) $\rightarrow$ ("Edges" low).

Finally, rule 12 was defined to avoid including isolated pixels in the output image, favouring only continuous lines. It also avoids including points by effect of noise, since this tends to generate isolated pixels in the image which represents the input's edges.

12. (DV (i, j + 1) low) AND (DH(i + 1, j) low) AND
    (DV (i, j - 1) low) AND (DH(i -1, j) low) $\rightarrow$ ("Edges" low).

(a)



(b)

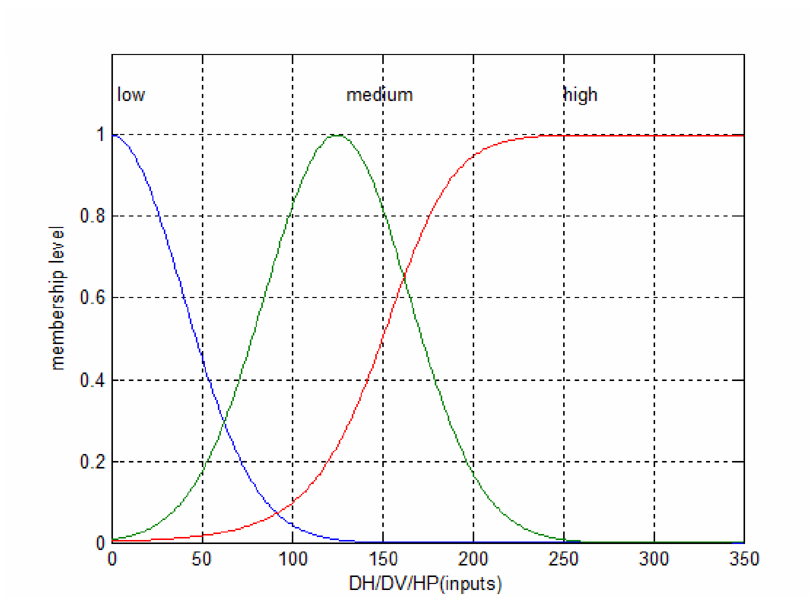


Figure 3.2 Membership function of fuzzy sets associated to (a) output E (edges) and input M and (b) to inputs *DH* , *DV* , *HP*

3.2 An Efficient Multilevel Fuzzy Edge Detector for Digital Images

The traditional fuzzy edge detection algorithm introduces the fuzzy enhancement method and is suitable for edge detection up to some extent [53]-[54]. The algorithm first enhances the image by means of mapping transformations, fuzzy enhancement operator, and inverse mapping transformation and then extracts the edge information from the enhanced image using “min” or “max” operator. This algorithm is computationally complex because the mapping transformation involves the exponential calculation and it will lead to loss of low intensity pixel.

Many improved algorithms have been proposed by various authors with simplified mapping transformation and optimized fuzzy enhancement operator [55],[56]. In this method the image is enhanced by dividing it into various levels and then edge detection is done by using two stages. The two- stage detection first determine the pixels which are potential edge candidate by means of local characteristic of the image and in the second step it determines true edges.

3.2.1 Overview of the fuzzy algorithm

Step 1. Computing the Threshold

The first step before fuzzification is image thresholding. Here thresholding is done by global thresholding [33] method. The reason for applying global thresholding as a method of thresholding in this case is its simple implementation. This is an iterative process given as follows

1. Select initial estimate for threshold T.
2. Segment the image into two groups *g1* and *g2*. Where *g1* is intensity values greater than or equal *T* and *g2* is intensity values less than *T*.
3. Compute a new threshold

   T = 0.5(mean (g1) +mean (g2));*
4. Repeat steps 2 through 3 until the difference in *T* in successive iterations is smaller than a predefined parameter *T0*.

Based on the threshold value, all the pixels in the image can be classified into two sets, namely *Fo* containing high gray level value greater than or equal to *T* and another *Fb* containing low gray level value less than threshold *T*. The mean value *Mo* for set *Fo* and *Mb* for set *Fb* can be computed as follows:

$$M_o = \frac{\sum_{f_{ij} \in F_o} f_{ij}}{S_o} \tag{3.1}$$

$$M_b = \frac{\sum_{f_{ij} \in F_b} f_{ij}}{S_b} \tag{3.2}$$

where *So* and *Sb* are sum of object pixels and the sum of background pixel.

Step 2. Computing the Fuzzy Membership value

The membership function as defined by Pal. King algorithm is given as:

$$\mu_{ij} = G(f_{ij}) = \left(1 + \frac{f_{\max} - f_{ij}}{F_d}\right)^{-F_e} \tag{3.3}$$

where *Fd* and *Fe* are reciprocal and exponential fuzzy factor respectively.

There is a large amount of calculation with exponential form for fuzzy membership function. Therefore the equation is redesigned as following:

$$\mu ij = G(f_{ij}) = \begin{cases} \dfrac{f_{ij} - f_{\min}}{M_o - f_{\min}} & f_{ij} \le M_b \\ \dfrac{M_o + M_b - 2f_{ij}}{M_o - M_b} & M_b < f_{ij} \le \dfrac{M_o + M_b}{2} \\ \dfrac{2f_{ij} - M_o - M_b}{M_o - M_b} & \dfrac{M_o + M_b}{2} < f_{ij} \le M_o \\ \dfrac{f_{\max} - f_{ij}}{f_{\max} - M_o} & f_{ij} > M_o \end{cases} \tag{3.4}$$

$i = 1, 2...., M; \quad j = 1, 2..., N$

where fmax and fmin denote the maximum and minimum gray value of image. M and N denote the rows and columns of the image respectively.

Step 3. Fuzzy Enhancement

After changing the image from spatial domain to fuzzy domain, the fuzzy enhancement operator Er is applied to get the enhanced image as follows:

$$\mu'_{ij} = E_r(\mu_{ij}) = E(E_{r-1}(\mu_{ij})) \tag{3.5}$$

$$E(\mu_{ij}) = \begin{cases} \dfrac{\mu_{ij}^{2}}{t} & 0 \le \mu_{ij} \le t \\ 1 - \dfrac{(1-\mu_{ij})^2}{1-t} & t < \mu_{ij} \le 1 \end{cases} \tag{3.6}$$

where r denotes the number of iterations, and to enhance the image moderately it is usually chosen as 2 or 3. *t* denotes fuzzy characteristic threshold, and its value can be chosen flexibly between 0 and 1. For the images considered here the results were mostly obtained for a t value varying in the range 0.5 to 1.

Step 4. Inverse transform of step 2

After enhancement in fuzzy domain, the inverse mapping is done to change the image from fuzzy domain into the spatial domain as follows:

$$h_{i,j} = G^{-1}(\mu'_{ij})$$
$$= \begin{cases} (M_b - f_{\min})\mu'_{ij} + f_{\min} & f_{ij} \le M_b \\ \dfrac{M_o + M_b - (M_o - M_b)\mu'_{ij}}{2} & M_b < f_{ij} \le \dfrac{M_o + M_b}{2} \\ \dfrac{(M_o - M_b)\mu'_{ij} + M_o + M_b}{2} & \dfrac{M_o + M_b}{2} < f_{ij} \le M_o \\ f_{\max} - (f_{\max} - M_o)\mu'_{ij} & f_{ij} > M_o \end{cases} \tag{3.7}$$

$i = 1, 2...., M; \quad j = 1, 2...., N$

3.2.2. Edge detection in two stages

1. *First Stage Edge Detection*: The first stage edge detections aim is to determine pixels which are probable edge candidates.

For any one pixel (i,j) with its gray value equal to fij, the 3x3 window centered around (i,j) is chosen. The mean value Mij of the gray values of all the pixels in the window is computed. The edge sign is determined according to relationship between Mij and fij as follows:

$$s_{ij} = \begin{cases} 1, & f_{ij} < M_{ij} \\ 0, & f_{ij} \ge M_{ij} \end{cases} \tag{3.8}$$

where sij=0 indicates that pixel(i,j) is not an edge pixel, while sij=1 indicates pixel (i,j) will be edge candidate.

2. *Second-Stage Edge Detection Operator*:

For the pixel (i,j) with sij=1,the 5x5 window centered around (i,j) is chosen, and it is divided into eight sub-windows as shown in Fig 1. Let the four pixels included in the lth (l=0….7) sub window be $(r_0^l, c_0^l), (i, j), (r_1^l, c_1^l)$ *and* (r_2^l, c_2^l)

The gradient values for the pixel (i,j) and the two neighbouring pixels in this sub window can be defined as follows:

$$d_{r_0^l,c_0^l} = abs(g_{i,j} - g_{r_0^l,c_0^l})$$

$$d_{i,j}^{l} = g_{r_1^l,c_1^l} - g_{i,j} \tag{3.9}$$

$$d_{r_1^l,c_1^l} = abs(g_{r_2^l,c_2^l} - g_{r_1^l,c_1^l})$$

$$\begin{array}{|ccccc|}\hline o&o&o&o&o\\ o&o&o&o&o\\ o&x&x&x&x\\ o&o&o&o&o\\ o&o&o&o&o\\ \hline\end{array}\quad
\begin{array}{|ccccc|}\hline o&o&o&o&x\\ o&o&o&x&o\\ o&o&x&o&o\\ o&x&o&o&o\\ o&o&o&o&o\\ \hline\end{array}\quad
\begin{array}{|ccccc|}\hline o&o&x&o&o\\ o&o&x&o&o\\ o&o&x&o&o\\ o&o&x&o&o\\ o&o&o&o&o\\ \hline\end{array}\quad
\begin{array}{|ccccc|}\hline x&o&o&o&o\\ o&x&o&o&o\\ o&o&x&o&o\\ o&o&o&x&o\\ o&o&o&o&o\\ \hline\end{array}$$

$$\begin{array}{|ccccc|}\hline o&o&o&o&o\\ o&o&o&o&o\\ x&x&x&x&o\\ o&o&o&o&o\\ o&o&o&o&o\\ \hline\end{array}\quad
\begin{array}{|ccccc|}\hline o&o&o&o&o\\ o&o&o&x&o\\ o&o&x&o&o\\ o&x&o&o&o\\ x&o&o&o&o\\ \hline\end{array}\quad
\begin{array}{|ccccc|}\hline o&o&o&o&o\\ o&o&x&o&o\\ o&o&x&o&o\\ o&o&x&o&o\\ o&o&x&o&o\\ \hline\end{array}\quad
\begin{array}{|ccccc|}\hline o&o&o&o&o\\ o&x&o&o&o\\ o&o&x&o&o\\ o&o&o&x&o\\ o&o&o&o&x\\ \hline\end{array}$$

Figure.3.6. The eight partition of the detection window

The additional edge sign for every pixel (i,j) in every sub window is determined as follows:

$$s_{ij}^{l} = \begin{cases} 1, & s_{ij} = 1 \ \ and \ \ d_{i,j}^{l} > d_{r_0^l,c_0^l} \ \ and \ d_{i,j}^{l} > d_{r_1^l,c_1^l} \\ 0, & otherwise \end{cases} \tag{3.10}$$

All the gradient values for pixel (i,j) with $s_{ij}^{l} = 1$ in the sub windows constitute

$$D_{ij} = \left\{ d_{ij}^{l} \mid s_{ij}^{l} = 1, l = 0.....7 \right\} \tag{3.11}$$

The maximal gradient value in D_{ij} will be used as the ultimate gradient value for the pixel (i,j) in the 5x5 detection window and the edge image will be produced when the gradient values of all the pixels in the enhanced image have been calculated as following:

$$d_{ij} = \begin{cases} \max(D_{ij}), & D_{ij} \neq \varnothing \\ 0, & D_{ij} = \varnothing \end{cases} \quad (3.12)$$

where $\varnothing$ denotes null set

3.3 Edge linking by morphological operators

The methods discussed in the previous section should result in pixels lying only on edges. However, practically this set of pixels seldom characterizes edge completely because of nose, breaks in the edge from nonuniform illumination, and other effect that introduce spurious intensity discontinuities. Thus edge detection algorithms are normally followed by edge linking procedures to bridge gaps in region boundary.

We apply simple morphological tools for the edge linking problem. The results of applying edge linking by morphological operators, on the edge detected image is shown in chapter 6 (Fig.6.15). The edge detection method considered for all these images is FMFED algorithm. The reason for not applying FIS based edge detector is its poor quality of edge detection compared to some older techniques like canny edge detector. The various morphological operators used for edge linking of these images are described below:

Cleaning – This operation removes isolated foreground pixels from the binary edge image.

Dilation – Dilation is an operation that “grows” or “thickens” objects in binary image. The specific manner an extent of this thickening is controlled by a shape referred to as a structuring element.

Mathematically, dilation is defined in terms of set operations. The dilation of A by B, denoted $A \oplus B$, is defined as

$A \oplus B = \{z \mid (\hat{B})_z \cap A \neq \varnothing |\}$

Where

$\varnothing$ is the empty set

B is the structuring element and

$\hat{B}$ is reflection of set B, defined as

$\hat{B} = \{w \mid w = -b, \ for \ b \in B\}$

Closing – Dilation and erosion are often applied to image in concatenation. Dilation followed by erosion is called a close operation. It is mathematically defined as

$f \bullet b = (f \oplus b) \odot b$

Where erosion is defined as a process that “shrinks” or “thins” objects in an binary image. The manner and the extent of shrinking is controlled by a structuring element. Mathematically, erosion is defined as

$A \odot B = \{z \mid (B)_z \cap A^c \neq \varnothing\}$

Where A^c is the complement of set A.

The simulation results and conclusion of the chapter are in chapter 6.

CHAPTER 4

Development of Algorithm for Segmentation of Color Images using Fuzzy Clustering

4

Preview

Advances in cognitive psychology over the past decades have revealed that visual data, in the form of scenes and pictures, are often mentally processed in visual terms alone, without any corresponding translation or recording into verbal labels or representation, and humans often respond strongly to color cues within image contents. In the past decade, color imaging and printing devices has become more affordable and computer power has been ever increasing. As a result color imaging has become very popular in many applications including object classification and recognition, video surveillance, image indexing and retrieval in image databases, feature based video compression, etc. In this chapter we discuss about color image segmentation, which is often a necessary computational process for color-based image retrieval and object recognition.

Image segmentation is a process of partitioning image pixels based on selected image features. The pixels that belong to the same region must be spatially connected and have the similar image features. If the selected segmentation feature is color, an image segmentation process would separate pixels that have distinct color feature into different regions, and simultaneously, group pixels that are spatially connected and have the similar color into the same region. Every pixel in the image must be assigned to a region when any segmentation algorithm terminates. In image processing two terms are usually seen very frequently close to each other: *clustering* and *segmentation*. When analyzing the color information of an image, for example and trying to separate regions or ranges of color components having same characteristics, the process is called *clustering*. Mapping the clusters onto the spatial domain and physically separating regions or surfaces in the image is called *segmentation*.

The objective of color clustering is to divide a color set into c homogeneous color clusters. Color clustering is used in a variety of applications, such as color image segmentation and recognition.

Color clustering is an inherently ambiguous task because color boundaries are often blurred. For example, consider the task of dividing a color image into color objects. In color images, the boundaries between objects are blurred and distorted due to the imaging acquisition process. Furthermore, object definitions are not always crisp, and knowledge about the objects in a scene may be vague. Fuzzy set theory and fuzzy logic are ideally suited to deal with such uncertainties. Fuzzy clustering models have proved a particularly promising solution to the color clustering problem. Such unsupervised models can be used with any number of features and clusters. In addition, they distribute membership values across the clusters based on natural groupings in feature space (Bezdek, 1999). In fuzzy clustering, the uncertainty inherent in a system is preserved as long as possible before decisions are made. Of the fuzzy clustering algorithms proposed to date, the fuzzy c-means (FCM) algorithm proposed by Bezdek is the most widely used in image segmentation because it has robust characteristics for ambiguity and can retain much more information than hard segmentation methods. Fuzzy c-means is an unsupervised technique that has been successfully applied to feature analysis, clustering, and classifier designs in fields such as astronomy, geology, medical imaging, target recognition, and image segmentation. An image can be represented in various feature spaces, and the FCM algorithm classifies the image by grouping similar data points in the feature space into clusters. This clustering is achieved by iteratively minimizing a cost function that is dependent on the distance of the pixels to the cluster centers in the feature domain.

4.1 Representation of Color Images

4.1.1 The colour data: vector representation

A. Bitmaps

The original and basic way of representing a digital colored image in a computer's memory is obviously a bitmap. A bitmap is constituted of *rows of pixels*, contraction of the words '*Picture Element*'. Each pixel has a particular value which determines it's appearing color. This value is qualified by three numbers giving the *decomposition of the color in the three primary colors* Red, Green and Blue. Any color visible to human eye can be represented this way. The decomposition of a color in the three primary colors is quantified by *a number between 0 and 255*. For example, white will be coded as R = 255, G = 255, B = 255; black will be known as (R,G,B) = (0,0,0); and say, bright pink will be : (255,0,255). In other words, an image is an enormous two dimensional array of color values, pixels, each of them coded on 3 bytes, representing the three primary colors. This allows the image to contain a total of 256x256x256 = 16.8 million different colors. This technique is also know as *RGB encoding*, and is specifically adapted to human vision. With cameras or other measure instruments we are capable of 'seeing' thousands of other 'colors', in which cases the RGB encoding is inappropriate. The range of 0-255 was agreed for two good reasons: The first is that the human eye is not sensible enough to make the difference between more than 256 levels of intensity (1/256 =0.39%) for a color. That is to say, an image presented to a human observer will not be improved byusing more than 256 levels of gray (256 shades of gray between black and white). Therefore 256 seems enough quality. The second reason for the value of 255 is obviously that it is *convenient for computer storage*. Indeed on a byte, which is the computer's memory unit, can be coded up to 256 values.

As opposed to the audio signal which is coded in the time domain, the image signal is coded in *a two dimensional spatial domain*. The raw image data is much more straight forward and easy to analyse than the temporal domain data of the audio signal. This is why we will be able to do lots of stuff and filters for images without transforming the source data, this would have been totally impossible for audio signal.

B. Vector representation of colors

As we have seen, in a bitmap, colors are coded on three bytes representing their decomposition on the three primary colours. It sounds obvious to a mathematician to immediately interpret *colors as vectors in a three dimension space where each axis stands for one of the primary colors.* Therefore we will benefit of most of the geometric mathematical concepts to deal with our colors, such as norms, scalar product, projection, rotation or distance. Figure 4.1, illustrates this new interpretation:

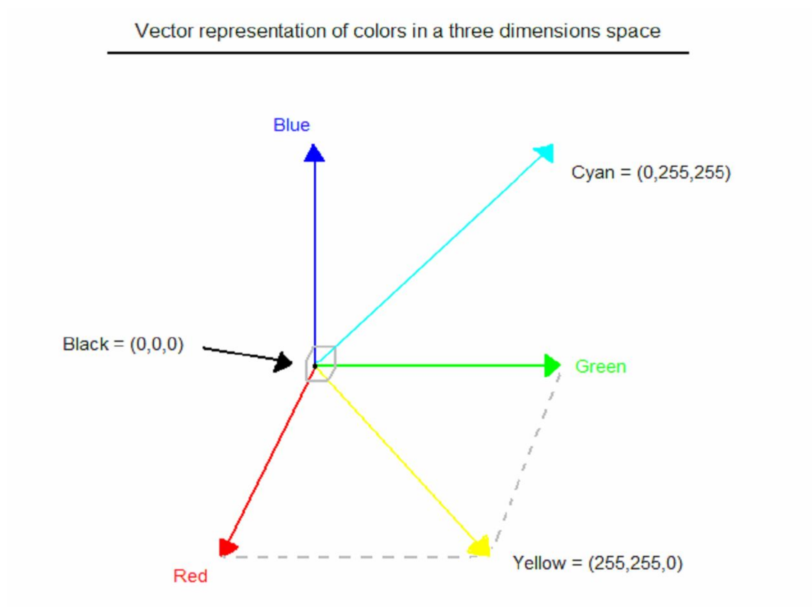


Figure 4.1. vector representation of color

4.2 Selection of Color Space

Sometimes it is necessary to adjust computer vision to human vision. Especially it is necessary when we are segmenting images, which were segmented by people and we try to replace people with computers or when we want to help people in segmentation of images. For this purpose we are using the L*a*b* color space. The L*a*b* color space consists of a luminosity layer 'L*', chromaticity-layer 'a*' indicating where color falls along the red-green axis and chromaticity-layer 'b*' indicating where the color falls along the blue-yellow axis. The non linear relationships for L* a* and b* are the same as for CIE XYZ (1931) and is another attempt to linearise the perceptibility of unit vector color differences. Again, it is in non-linear, and the conversions are still reversible. Colouring information is referred to the color of the white point of the system. The non linear relationships for L* a* and b* are the same as for CIELUV and are intended to mimic the logarithmic response of the eye.

The color space used in the initialization is of great importance because the shapes and distribution of clusters depend on the color space (Tominaga, 1992). Typically, raw color data are expressed in the RGB color space. However, RGB is not a perceptually uniform space. The CIELAB color space, adopted as an international standard in the 1970's, provides perceptually uniform space, which means the Euclidean distance between two color points in the CIELAB color space corresponds to the perceptual difference between the two colors by the human vision system (Wyszecki and Stiles, 2000). This property has made the CIELAB color space to be attractive and useful for color analysis, and the CIELAB color space has shown its superior performance than other color spaces in many color image applications (Paschos, 2001; Gong et al., 1998; Chang and Wang, 1996; Li and Yuen, 2000; Shafarenko et al., 1998). Based on these reports, the CIELAB color apace has been chosen for color clustering. The transformation from RGB to CIELAB is performed as followed.

The L parameter has a good correlation with perceived lightness. The LAB cube root color coordinate system was developed to provide a computationally simple measure of color in agreement with Munsell color system[58]. The color coordinates are

$$L = 25\left[100\frac{Y}{Y_o}\right]^{1/3} - 16 \tag{4.1}$$

$$A = 500\left[\left[\frac{X}{X_o}\right]^{1/3} - \left[\frac{Y}{Y_o}\right]^{1/3}\right] \tag{4.2}$$

$$B = 200\left[\left[\frac{X}{X_o}\right] - \left[\frac{Z}{Z_o}\right]^{1/3}\right] \tag{4.3}$$

where Xo, Yo, Zo are the tristimulus values for the reference white and X, Y, Z are the tristimulus value of the image pixels. We approximate these tristimulus values from (RGB) by the linear transformation:

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} 0.607 & 0.174 & 0.200 \\ 0.299 & 0.587 & 0.114 \\ 0.000 & 0.066 & 1.116 \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix} \tag{4.4}$$

The reference white is (Ro,Go,Bo) = (255,255,255).

Basically, 'L' is correlated with brightness, 'A' approximates redness - greenness, and 'B' with yellow – blueness. These coordinates are used to construct a Cartesian color space where the Euclidean distance is used that is,

$$\Delta E^*_{ab} = \sqrt{\Delta L^{*2} + \Delta a^{*2} + \Delta b^{*2}} \tag{4.5}$$

4.3 Fuzzy c-means Algorithm

Clustering is a process for classifying objects or patterns in such a way that samples of the same group are more similar to one another than samples belonging to different groups. Many clustering strategies have been used, such as the hard clustering scheme and the fuzzy clustering scheme, each of which has its own special characteristics. The conventional hard clustering method restricts each point of the data set to exclusively just one cluster. As a consequence, with this approach the segmentation results are often very crisp, i.e., each pixel of the image belong to exactly just one class. However, in many real situations, for images, issues such as limited spatial resolution, poor contrast, overlapping intensities, noise and intensity inhomogeneities variations make this hard (crisp) segmentation a difficult task. Due to this fuzzy set theory was proposed, which produced the idea of partial membership of belonging described by a membership function.

Fuzzy clustering as a soft segmentation method has been widely studied and successfully applied to image segmentation [59-63].The fuzzy c-means (FCM) algorithm, proposed by Dunn and generalized by Bezdek[64], has the function to describe the fuzzy classification for the pixels by calculating the fuzzy membership value. Fuzzy c-means algorithm is a data clustering algorithm in which each data point belongs to a cluster to a degree specified by a membership grade. It minimizes an objective function, with respect to fuzzy membership U, and set of cluster centroids V

$$J(\mathrm{U},\mathrm{V}) = \sum_{k=1}^{n} \sum_{i=1}^{c} (u_{ik})^m d^2(x_k, v_i) \tag{4.6}$$

where

$$X = \{x_1, x_2, \ldots, x_n\} \subseteq R^p$$

c - the number of cluster centers or data subsets

m - the weighting exponents, 1 for 'hard' clustering, and increasing for fuzzier clustering;

$d^2(x_k, v_i)$ - the distance measure between object xk and cluster center vi;

n - the total number of pixels in image;

uik - the fuzzy membership value of pixel k in cluster i;

vi - the cluster center for subset i in feature space;

U – the fuzzy c-partition

The above fuzzy c-mean algorithm uses iterative operation to get U and V and finally minimizes the objective function. The algorithm is achieved as following:

1. Fix the number of cluster c, $2<c<n$;

   Fix m, $1<m<\infty$

2. Initialize the fuzzy c-partition $\mathrm{U}^{[0]}$;
3. Assume the steps b = 1, 2 ,…. ;
4. Calculate the c cluster centers $\{\mathrm{V}_i^{(b)}\}$ with $\mathrm{U}^{(b)}$, the cluster center for cluster i is .

$$\mathrm{v}_i = \frac{\sum_{k=1}^{n} (\mathrm{u}_{ik})^m x_k}{\sum_{k=1}^{n} (\mathrm{u}_{ik})^m} \tag{4.7}$$

5. Update $U^{(b)}$, calculate the membership $U^{(b+1)}$:

(a) Calculate Ik and Tk

$I_k = \{i \quad 1<i<c\};$

$d_{ik} = abs(x_i - v_k) = 0;$

$T_k = \{1,2, ...,c\} - I_k;$

(b) For data set k, calculate the new membership values:

(i) if $I_k = 0$

$$u_{ik} = \frac{1}{\sum_{j=1}^{c}\left(\frac{d_{ik}}{d_{jk}}\right)^{\frac{2}{m-1}}} \tag{4.8}$$

(ii) else

$u_{ik} = 0, \ \forall\ i \in T_k$ and

$\sum_{i\in I_k} u_{I_k} = 1$

6. Compare $U^{(b)}$ and $U^{(b+1)}$ in a convenient matrix norm,

If $U^{(b)} - U^{(b+1)} < \in_L$, stop;

Otherwise, set b = b+1 and go to step 4.

Here $U^{(0)}$ is the initial partition and can be randomly set or by an approximation method. $\in_L$ is the convergence threshold. The introduction of the term m makes the segmentation flexible, m = 1 for 'hard' clustering. The increase of the values of m stresses the fuzzy properties. The FCM process is guaranteed to converge for m >1.

4.4 Segmentation Method:

The Segmentation process consists of several steps. The first step is the conversion of the input image to chosen feature space, which may depend on the clustering method used. In our case the input image is converted from RGB colour space to LAB colour space. The L, A and B values are used as input to the clustering method.

Next step after the conversion of input image color is the application of clustering algorithm. In our case we use fuzzy c- mean clustering as described in the section above.

After these two steps the segmentation process is followed as described:

Assumptions: Image transformed into feature space, number of clusters is c, stop condition is $\in$, fuzziness parameter m = 2.

Step 1: Convert the given RGB image into desired feature space (LAB colour space in this case).

Step 2: Cluster image in feature space, with given conditions: number of clusters is c, fuzziness index is m=2 and stop condition is $\in$.

Step 3: The FCM iteration is stopped when the maximum difference between two objective functions at two successive iterations is less than or equal to that of a fixed value.

Step 4: For every pixel I(r,c) of image I, where ‘r’ is number of row and ‘ c’ is number of column, the following steps are followed.

Step 4.1: All the pixels were considered belonging to one of the predetermined regions or clusters. The number of cluster should be chosen carefully.

Step 4.2: The defuzzification process [23] takes place in order to convert the fuzzy partition matrix U to crisp partition. A number of methods have been developed to defuzzify the partition matrix, among which the maximum membership procedure is the most important. The procedure assigns the object k to the class c with the highest membership

$$c_k = \arg_i\{\max(u_{ik})\} \tag{4.9}$$

Step 4.3: The decision on how to assign the pixel I(i,j) to various clusters was based on wining u_{ik} having highest value among the clusters.

Step 5: The pixel I(i,j) would be painted the same colour as the cluster to which it belongs the most.

4.5 Image segmentation under uneven illumination of objects

Image degradation is inevitable during the transmission and conversion of images. For example, the quality of an image shot by a camera is sometimes low due to the distortion of camera's optics system, low light conditions, the relative motion of the photographed object and the camera, the environmental change and the random disturbance. If we perform clustering operation on such images we are likely to get wrong classification of objects present in the image due to distortion of the image. Hence an enhancement operation has to be carried out as a preprocessing step on such images before clustering operation is performed on them. The enhanced image avoids wrong classification to great extent. The image enhancement is an important technique that can improve the quality of the degraded image and provide some interested image features selectively. Image enhancement algorithms have been designed to process a given image so the results are better than the original image for their applications.

When the objective is to improve perceptual aspects, desirable image enhancement can be performed by the contrast and dynamic range modification.

Processing techniques for image enhancement can be classified into spatially uniform operators and spatially non-uniform operators. Linear contrast stretch, histogram equalization are two of the most widely used spatially uniform technique. Adaptive histogram-equalization (AHE) [67], contrast-limited adaptive histogram equalization (CLAHE) [68] belongs to the second class of image-contrast enhancement methods. While the spatially uniform methods use a transformation applied to all the pixels of the image, the later methods use an input–output transformation that varies adaptively with the local characteristics of the image. Spatially non-uniform operators usually provide a better performance than spatially uniform operators. The linear contrast-stretch method can hardly enhance all parts of the image simultaneously. Histogram equalization tends to over-enhance the image contrast if there are high peaks in the histogram. Adaptive histogram equalization applies locally varying gray-scale transformation each small region (block) of the image, thus requiring the determination of the block size. An improvement on this technique is represented by the CLAHE method. In contrast-limited adaptive histogram equalization, the local contrast-gain is limited by restricting the height of local histograms. This method provides for local enhancement of region in an image. It reduces undesired noise amplification and reduces boundary artefacts.

4.5.1. Contrast limited adaptive histogram equalization

Contrast Limited Adaptive Histogram Equalization (CLAHE) is an extension to Adaptive Histogram Equalization (AHE) which limits the maximum contrast adjustment that can be made to any local histogram. This limitation is useful so that the resulting image does not become too noisy (which is a problem with AHE). The limitation is performed by allowing a set maximum number of pixels within each gray level associated with a local histogram. After clipping the histogram, the pixels that were clipped are equally redistributed over the whole histogram to keep the whole histogram count unchanged. It operates on small data regions (tiles) rather than the entire image. Each tiles contrast is enhanced so that the histogram of each output region approximately matches the specified histogram (uniform distribution in this case).

4.5.2 Segmentation method

The algorithm developed is a contrast limited adaptive histogram equalization based FCM. Hence, it is called *CLAHEFCM*. The segmentation process consists of several steps. The first step is the conversion of the input image to chosen feature space, which may depend on the clustering method used. In our case the input image is converted from RGB color space to LAB color space. The L, A and B values are used as input to the clustering method.

Next step after the conversion of input image color space is the application of enhancement method, followed by clustering algorithm. In our case we use fuzzy c- mean clustering as described in the section above.

After these two steps the segmentation process is followed as described:

Assumptions: Image transformed into feature space, number of clusters is c, stop condition is $\in$, fuzziness parameter m = 2.

Step 1: Covert the given RGB image into desired feature space (LAB color space in this case).

Step 2: Next we normalize the brightness layer L, by dividing by 100. After that we apply the contrast limited adaptive histogram enhancement (CLAHE) algorithm to the luminosity layer.

Rest of the steps are same as the step 3- step 5 in section 4.4.

The simulation results and conclusion of the chapter are in chapter 6.

CHAPTER 5

Development of Algorithm for Segmentation by Incorporating Spatial Property of Pixels in Fuzzy Clustering

5

Preview

Fuzzy c-means clustering is an unsupervised technique that has been successfully applied to feature analysis, clustering, and classifier designs in fields such as astronomy, geology, medical imaging, target recognition, and image segmentation. An image can be represented in various feature spaces, and the FCM algorithm classifies the image by grouping similar data points in the feature space into clusters. This clustering is achieved by iteratively minimizing a cost function that is dependent on the distance of the pixels to the cluster centers in the feature domain.

The pixels on an image are highly correlated, i.e. the pixels in the immediate neighbourhood posses nearly the same feature data. Therefore, the spatial relationship of neighbouring pixels is an important characteristic that can be of great aid in imaging segmentation. General boundary detection techniques have taken advantage of this spatial information for image segmentation. However, the conventional FCM does not fully utilize this spatial information.

5.1 FCM-Related Extensions

The most direct way to compensate for the drawback of FCM is to smooth the image before segmentation. However, standard smoothing filters lead to a loss of important image details. Various extensions of the FCM algorithm with attempt to accommodate noise have been presented by many researchers. Tolias and Panas post-processed the membership function to smooth the noise effect [69]. Acton and Mukherjee incorporated multiscale information to enforce spatial constraints [70].

The most popular approach for increasing the robustness of FCM to noise is to modify the objective function directly. Dave proposed the idea of a noise cluster to deal with noisy clustering data in the approach known as NC [71]. Noise is effectively clustered into a separate cluster which is unique from from signal clusters. However, it is not suitable for image segmentation, since noisy pixels should not be separated from other pixels, but assigned to the most appropriate clusters in order to reduce the effect of noise.

Another similar method, developed by Krishnapuram and Keller [72], is called possibilistic c-means (PCM), which interprets clustering as a possibilistic partition. Instead of having one term int the objective function, a second term is included, forcing the membership to be as high as possible without a maximum limit constraint of one. However, it caused clustering being stuck in one or two clusters.

Pedrycz and Waleztzky [73] took advantage of the available classified information and actively applied it as a part of their optimization procedures. Ahmed et al. [22] modified the objective function of the standard FCM by introducing a term that allowed the labelling of a pixel to be influenced by the labels in its immediate neighbourhood. Zhang Yang , Fu-lai Chuang et al.[75] developed a robust fuzzy clustering- based segmentation method for noisy images. A robust modified FCM in the sense of a novel objective function is derived. The applicability of the proposed modified FCM is also explored.

Jiayin Kang et al.[25] proposed another such modified FCM where objective function was modified by incorporating the spatial neighbourhood information into the standard FCM algorithm. Y. Yang et al. proposed a novel penalized fuzzy c-means (PFCM) algorithm for image segmentation, the penalty term acts as a regularizer in the algorithm which is inspired by neighbourhood maximization (NEM) algorithm and is modified in order to satisfy criterion of FCM algorithm [23].

S.Shen,W.Sandham et al. [24] presented an algorithm called IFCM. A neighbourhood attraction, which is dependent on relative location and features of neighbouring pixels, is used to improve the segmentation results. This method changed the distance function used in FCM which is the distance between pixel intensity and the cluster intensities and a neural network optimization technique was used to adjust parameters in the modified distance function. But problem with this method is that it requires an extra neural network optimization step for adjusting parameters of the distance function. Hence, this makes the algorithm complex. Keh-Shih Chuang, Hong-Long Tzeng, et al. [74] presented a fuzzy c-means (FCM) algorithm that incorporated spatial information into the membership function for clustering, and the membership weighting of each cluster is altered after the cluster distribution in the neighbourhood is considered. The problem with this method is that it does not produce smooth edges.

All these methods except the last two methods inevitably introduce computation issues, by modifying most equations along with the modification of the objective function, and have to lose the continuity from FCM, which is well-realized with many types of software, such as MATLAB.

5.2 Development of algorithm for incorporating spatial relationship of neighbouring pixels into FCM

5.2.1 Method

This proposed method is based on the FCM incorporating *spatial function* [74] proposed by K-S Chuang et al. One of the important characteristics of an image is that its neighbouring pixels are highly correlated to each other. The probability that a pixel neighbourhood will belong to same cluster is very high. This property of the pixels is quite helpful when the image is affected by noise. As the spatial relationship among pixels is not considered in the standard FCM algorithm a spatial function is introduced to take into account the neighborhood property.

For finding the spatial function, the membership information of each pixel of a cluster is converted to its spatial domain to get the complete image. Then we calculate the *spatial function*, using the following definition

$$s_{ik} = \sum_{k \in NB(x_k)}^{M} u_{ij} \tag{5.1}$$

where $NB(x_k)$ represents a square window centered on pixel x_k (1<k<n, where n is the total number of pixels in the image) in the spatial domain image containing the membership information of each pixel to a particular cluster 'i '. A 5x5 window was used for this work. Just like the membership function u_{ij} the spatial function s_{ik} gives the membership of the kth pixel to a particular cluster 'i '.

The spatial function is modified in order to take into account the properties of a local neighborhood in a way that the membership of each pixel results as a weighted sum of the pixels in the 5x5 neighborhood. This enables smoothening of the edges or boundaries of objects present in an image. Assuming M as the 5x5 neighborhood of the pixel j, the membership function to a cluster i is modified as follows:

$$h_{ik} = \frac{(h_{ik} + s_{ik})}{25} \tag{5.2}$$

Hence the new algorithm developed is named *Modified spatial fuzzy c means* (MSFCM)

The spatial function is then introduced in the membership function as follows:

$$u'_{ik} = \frac{u_{ik}^{p} h_{ik}^{q}}{\sum_{j=1}^{c} u_{jk}^{p} h_{jk}^{q}} \tag{5.3}$$

where p and q are parameters which control the relative importance of both functions. If the pixels in an image are not affected by noise then spatial function will only fortify the original membership, and the clustering result remains unchanged. However, for a noisy pixel, this formula reduces the weight of a noisy cluster by the labels of its neighboring pixels. As a result, misclassified pixels from noisy regions or spurious blobs can easily be corrected.

The clustering is a two-pass process. In the first pass we use the standard FCM to calculate the membership value for each pixel. The membership value for each pixel to different clusters is then mapped to spatial domain and the *spatial function* is calculated from that.

In the second pass, the FCM iteration proceeds with the new membership function that is incorporated with the spatial function. The iteration of spatial FCM algorithm stopped when the difference between the present and the previous objective function is less than or equal to a certain value (10^{-5}).

After the convergence, defuzzification is applied to assign each pixel to a specific cluster for which the membership is maximal.

5.2.2 Segmentation method

The Segmentation process consists of several steps. The various steps involved in the method are shown in Fig. 5.1. The first step is the conversion of the input image to chosen feature space, which may depend on the clustering method used. In our case the input image is converted from RGB color space to LAB color space. The L, A and B values are used as input to the clustering method.. In our case we use fuzzy c- mean clustering as described in the section above.

After these two steps the segmentation process is followed as described:

Assumptions: Image transformed into feature space, number of clusters is c, stop condition is $\in$, fuzziness parameter m = 2.

Step 1: Convert the given RGB image into desired feature space (LAB color space in this case).

Step 2: Cluster image in feature space, with next conditions: number of clusters is c, fuzziness index is m=2 and stop condition is $\in$.

Step 3: The membership information of each pixel is mapped to the spatial domain, and the spatial function is calculated.

Step 4: The new membership function is calculated using equation (4).

Step 5: The FCM iteration proceeds with the new membership function. The iteration is stopped when the maximum difference between two objective functions, at two successive iterations, is less that a fixed value.

The next steps are same as steps 4-5 in segmentation method of section 4.4

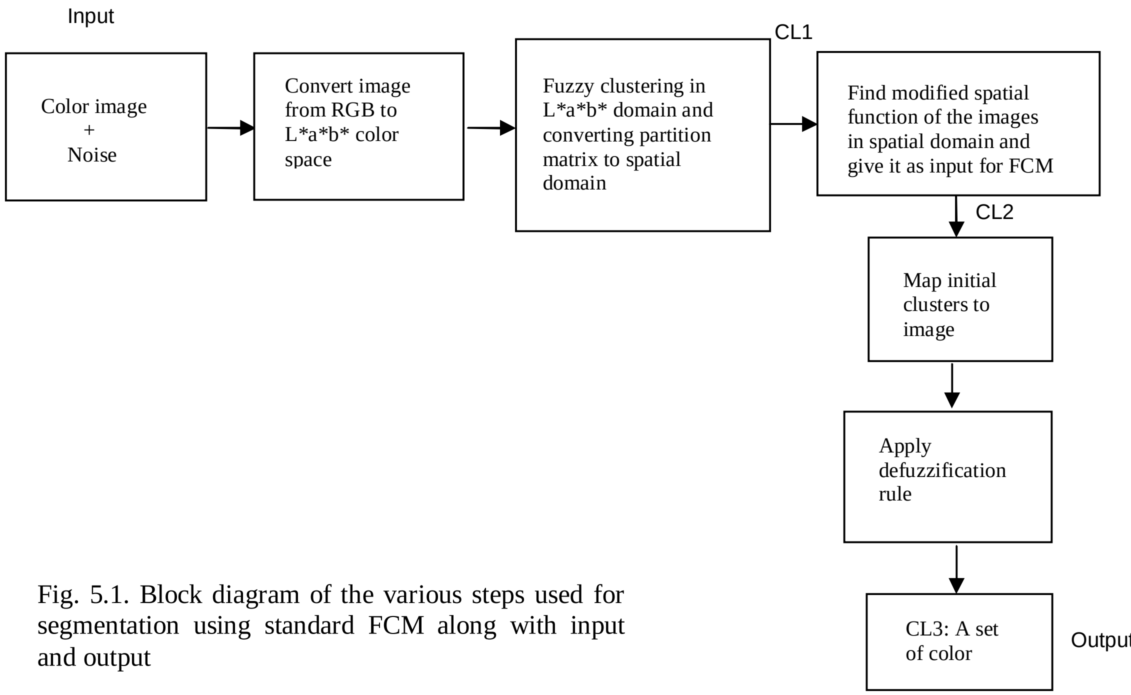
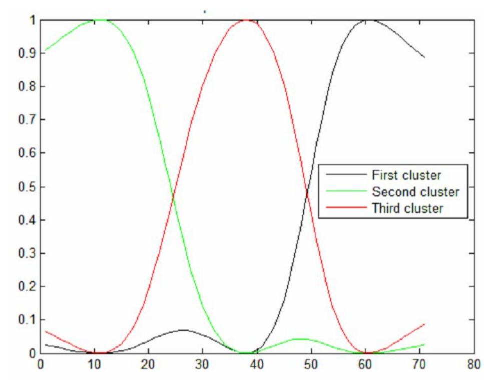
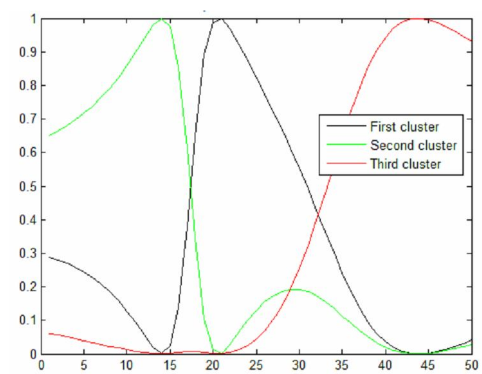


Fig. 5.1. Block diagram of the various steps used for segmentation using standard FCM along with input and output

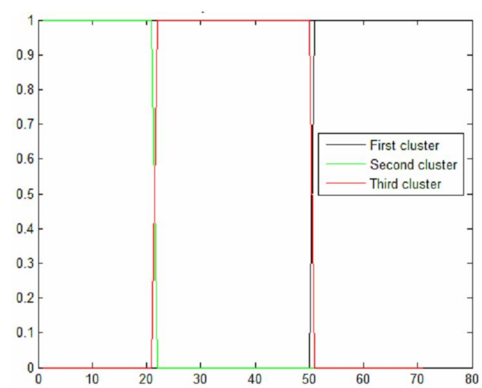
The ‘hand’ image was divided into three clusters, the three clusters consists of the hand, green ring and the background. The membership function of each of these three clusters with respect to ‘A’and ‘B’ values of image pixels, calculated by standard FCM and proposed method called *modified spatial FCM (MSFCM)* is shown in Fig. 5.2.The membership function for both sFCM and proposed method are same.



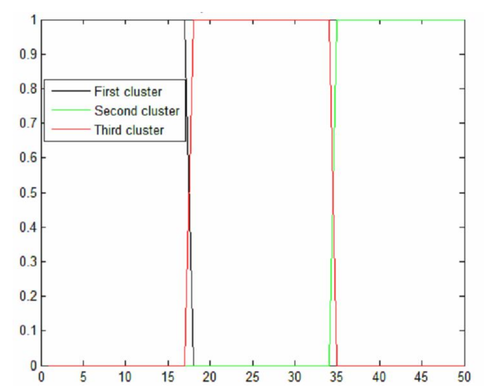
(a)



(b)



(c)



(d)

Fig. 5.2. (a) Membership function of first, second and third cluster with respect to a* values of image pixels using FCM. (b) Membership function of first, second and third cluster with respect to b* values of image pixels using FCM. (c) Membership function of first, second and third cluster with respect to b* values of image pixels using MSFCM. (d) Membership function

5.3 Segmentation of noisy colour images using neighbourhood property of a digital image

5.3.1 Method

A new algorithm based on the IFCM (Improved Fuzzy c- means) [24] neighbourhood attraction is proposed. The algorithm does not change the distance function of the FCM, hence avoiding an extra neural network optimization step for the adjusting parameters of the distance function; it is called *Neighbourhood Attraction FCM (NAFCM)*. During clustering, each pixel attempts to attract its neighbouring pixels towards its own cluster. This neighbourhood attraction depends on two factors: the pixel intensities or feature attraction, and the spatial position of the neighbours or distance attraction, which also depends on neighbourhood structure.

The first parameter, feature attraction, is given by the function

$$H_{ij} = \frac{\sum_{k=1}^{S} u_{ik} g_{jk}}{\sum_{k=1}^{S} g_{jk}} \tag{5.4}$$

Where g_{jk} is the intensity difference between study pixel j and its neighbour pixel k.

$$g_{jk} = \left| x_j - x_k \right|$$

u_{ik} is the membership of the neighboring pixel k to the ith cluster, and S is the number of neighboring pixels.

The distance attraction function is given by

$$F_{ij} = \frac{\sum_{k=1}^{S} u_{ik}^2 q_{jk}^2}{\sum_{k=1}^{S} q_{jk}^2} \tag{5.5}$$

The neighbourhood structure is of the form

$$K_j = \left\{ k \in N \mid 0 < (a_j - a_k)^2 + (b_j - b_k)^2 \le Q \right\} \tag{5.6}$$

Where $(a_j, b_j), (a_k, b_k)$ denote the coordinates of the pixel j, k. Q is a constant, equal to $2^{(L-1)}$, and L is the level of the neighbourhood. Fig.5.7 shows the neighbourhood for different levels. We consider L =2.

q_{jk} in (5) can be described as follows:

$$q_{jk} = (a_j - a_k)^2 + (b_j - b_k)^2 \tag{5.7}$$

After getting the functions Hij and Fij each of these matrixes are converted into spatial domain and perform the smoothing operation on them using averaging filters. This operation is done in order to reduce the effect of noise in the image.

$$w_{ij} = \sum_{i=1}^{M} \sum_{j=1}^{N} H_{ij}$$

$$h_{ij} = (h_{ij} + w_{ij}) / 25 \tag{5.8}$$

$$m_{ij} = \sum_{i=1}^{M} \sum_{j=1}^{N} F_{ij}$$

$$f_{ij} = (f_{ij} + m_{ij}) / 25 \tag{5.9}$$

h_{ij} and f_{ij} are given as input to the FCM algorithm. We take the number of cluster according to object of interest for a particular problem.

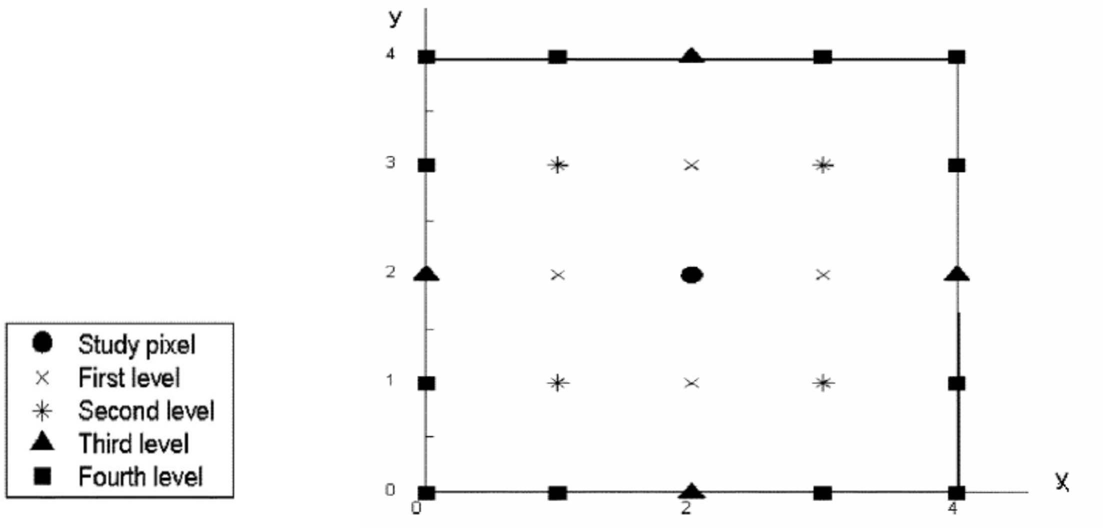


Figure. 5.7. Neighborhood structure definitions. (A higher level includes pixels labeled as the number of the level and pixels in all lower levels).

5.3.2 Segmentation method

The Segmentation process consists of several steps. The first step is the conversion of the input image to LAB color space. Next step is finding the two attraction features for the image, followed by clustering algorithm. In our case we use fuzzy c- mean clustering as described in the section above.

After these two steps the segmentation process is followed as described:

Assumptions: Image transformed into feature space, number of clusters is c, stop condition is $\in$, fuzziness parameter m = 2.

Step 1: Take the desired noisy colour image (Skin tumor images in this case) and convert the image to LAB color space.

Step 2: Find the feature attraction and distance attraction function as defined by equation (5.4) and (5.5).

Step 3: Convert the feature attraction and distance information into spatial domain, and perform smoothing operation using averaging filters on the image matrix formed in spatial domain. The smoothed images formed from the two matrixes are found by equation (5.8) and (5.9).

Step 4: The data from equation (5.8) and (5.9) are used as input for the FCM algorithm.

Step 5: The FCM iteration is stopped when the maximum difference between two objective functions at two successive iterations is less than or equal to that of a fixed value.

Rest of the steps are same as steps 4-5 in section 4.4.

5.4 Segmentation by using morphological operator

Apart from the segmentation method described in previous section for segmenting tumor images, morphological operators can also be used for segmentation. The results of applying morphological operators for segmentation after clustering is shown in figure 6.40- figure 6.41. The steps involved for segmentation by using morphological operators are as follows.

Step 1: First the cluster images are converted to black and white images by thresholding and then the regions in the images are filled using morphological tool

Step 2: Fill the holes (hole is an area of dark pixels surrounded by lighter pixels) inside the region of the black and white cluster images.

Step 3: Perform morphological opening on the image with a structuring element (square mask of size 3x3). This is done to smooth the contour of the object, break narrow isthumuses and eliminates thin protrusions.

Step 4: Remove from the binary segmented imgaes all connected components (objects) that have less than 200 pixels. We select 200 pixels as it is sufficient to remove object which do not belong to region of interest, i.e the affected region.

Step 5: Finally find the perimeter of the objects in binary image and overlay them on the original image.

5.5 Application of NAFCM algorithm in segmentation of melanoma images:

Segmentation of Melanoma images using the above algorithm:

Melanoma, the most serious type of skin cancer, develops in the cells that produce melanin — the pigment that gives the skin its color. Melanoma can also form in eyes and, rarely, in internal organs, such as intestines.

The exact cause of all melanomas isn't clear, but exposure to ultraviolet (UV) radiation from sunlight or tanning lamps and beds greatly increase risk of developing melanoma.

Avoiding excessive sun exposure can prevent many melanomas. And making sure you know the warning signs of skin cancer can help ensure that cancerous changes are detected and treated before they have a chance to spread. Melanoma can be successfully treated if it is caught in early stages.

The first melanoma symptoms often are: a change in an existing mole, or the development of a new, unusual-looking growth on the skin. But melanoma can also occur in otherwise normal looking skin.

Unusual moles that may indicate melanoma:

Characteristics of unusual mole that may indicate melanoma or other skin cancer follow the A-B-C-D guide developed by the American Academy of Dermatology:

- A is for asymmetrical shape. Look for moles of irregular shapes, such as two very different-looking halves.
- B is for irregular border. Look for moles irregular, notched or scalloped borders – the characteristics of melanoma.
- C is for changes in color. Look for growths that have many colors or an uneven distribution of color.
- D is for diameter. Look for new growth in a mole larger than about ¼ inch (6 mm.).

For every symptom listed above, we take one example into consideration. The examination of growth on skin is done automatically by use of the proposed NAFCM algorithm. The simulation results are present in Fig. 6.32-Fig. 6.35.

The simulation results and conclusion of the chapter are in chapter 6.

CHAPTER 6

Simulation Results and Discussion

6

Preview

The simulation results of all the chapters and their conclusion is presented in the chapter. The image metrics like partition coefficient *Vpc*, partition entropy *Vpe* and the percentage of misclassified pixels are used in the chapter to compare between the various existing and proposed algorithms. Extensive qualitative and quantitative analysis is done for comparing the clustering and segmentation results obtained using the different algorithms, under increasing noise condition. The algorithms are tested on synthetic image, real world image and biomedical image.

6.1 Simulation Results

The algorithms are implemented on Matlab 7.0 (The Mathworks Inc.). The processesor used is a Pentium IV core 2 duo processor, 2.4Ghz (clock), 2GB (RAM), Windows vista 64 bit operating system.

Fig. 6.1(a) is an infrared image of an aluminium weld with porosity or crack. Fig 6.1(b)-(d) shows the result of applying region growing for segmentation of the crack in the weld.

Fig. 6.2 shows a scenery image and the result of applying region growing for segmenting a particular field region. Figure 6.3 shows the result of applying region split and merge on the weld crack image of Fig. 6.1(a). Fig. 6.4(a) shows image of Saturn planet with a distant isolated star near its right bottom side. Fig. 6.4(b) shows the result of applying a point detector mask shown in Fig.2.3. Fig. 6.5(a) shows a pill set image. Fig. 6.5(b)-(c) is the result of running a horizontal mask (Fig.2.4) on the image and thresholding it. Fig 6.6 (a) shows lena image and Fig.6.6(b)-(f) shows the result of applying Roberts, Prewitt’s, Sobel, LoG and Canny edge detector.

The FIS based edge detection described in section 3.1 is tested on different images, its performance being compared to that of the other derivative based popular edge detectors like, Sobel operator and Canny edge detector. Fig. 6.7(a) shows a block image which has varying gray levels on its two faces that are visible. Fig.6.7 (b)-(d) shows the result of applying Sobel and Canny edge detector and FIS system respectively. Fig, 6.8(a) depicts the image of digital cameras calibration pattern, in which there is a high contrast variation. Fig.6.8 (b)-(d) shows the result of applying Sobel and Canny edge detector and FIS system respectively. Fig. 6.9(a) shows a $230 \times 325 \times 8$ bits standard image that is used for the calibration in the visual system. Fig.6.9 (b)-(d) shows the result of applying Sobel and Canny edge detector and FIS system respectively. The edge detection performance of the methods Sobel operator, Canny edge detector and Fuzzy Inference system are compared in terms of the image quality.

The FMFED algorithm described above has been tested on some test images and its qualitative performance is compared to two popular edge detectors – Sobel and Canny edge detectors [57]. The fuzzy enhancement operator is tuned to allow good results while extracting edges of the image. For the images considered here the value of the fuzzy enhancement operator is mostly varied between the ranges 0.5 to 1. The first test image (Fig. 6.10) considered for comparison of the simulation results is a bird image, the second image (Fig. 6.11) is a tire image where the object of interest and the background have same gray level values. The third test image (Fig. 6.12) is a MRI image and the fourth (Fig. 6.13) and fifth (Fig. 6.14) test images are X-ray images. The qualitative comparison between Sobel edge detector, Canny edge detector and the FIS algorithm on the different test images is shown in Fig. 6.10(b)-(d) to Fig. 6.14 (b)-(d).

The results of applying morphological operators described in the section above is shown in Fig. 6.15 .The figure shows three edge detected image by applying FMFED algorithm in Fig. 6.15(a)-(c). The result of applying morphological operators is shown in figure 6.15(d)-(f). By observing the results it can be said that morphological operators are not the best way to fill the gaps in the edge images and hence there is scope for applying some other edge linking algorithm.

The FCM based segmentation described in chapter 4, section 4.4 has been tested on some colour test images in LAB and their results are shown. The number of cluster was chosen in such a manner that we are able to segment the region of interest based on color completely from the image provided. Selecting large values for number of cluster m, would lead to not so good generalization of the image. If too low values for the number of cluster are selected, the neighbourhood colours may be confused. Fig. 6.16 and Fig. 6.17, show comparison between segmentation in RGB color space and LAB color space. The quantitative comparison between the two color spaces is done in Table 6.1 and Table 6.2. The effect on partition coefficient V_{pc} and partition entropy V_{pe} for increasing noise has been studied in these tables.

The images considered as test images for applying FCM algorithm are shown in Fig.6.18 (a) – Fig.6.21 (a). Fig.6 .18(a) is an image of some vegetation in desert area. Fig 6.18(b)-(d) shows the two clusters formed by using c=2 and the segmented image respectively. Fig. 6.19(a) is the image of an woman. Fig 6.19(b)-(e) shows the three clusters formed by choosing selecting c = 3 and the segmented image. Fig. 6.20(a) is a biomedical image showing hyper pigmentation of skin of an old lady. The cheek region shows dark patches due to hyperpigmentation. Fig 6.20(b)-(d) shows the two clusters formed by selecting c=2 and the segmented image. Fig. 6.21(a) is an image containing a bird flying in sky. Fig 6.21(b)-(d) shows the two clusters formed by selecting c=2 and the segmented image.

The CLAHE based FCM algorithm described in section 4.6 has been tested on some color test images in LAB color space and their results are shown in Fig 6.22-6.25. Fig.

6.22(a) is a close aerial view of a landscape, which has a water body along with some dry areas. Fig 6.22(b) shows the enhanced image. Fig 6.22(c)-(d) shows the segmentation results for Fig. 6.22(a) and Fig. 6.22(b) respectively. Fig. 6.23(a) is the image of a person’s cheek region suffering from hyper-pigmentation. Fig. 6.23(b) is the enhanced image. Fig 6.23(c)-(d) shows the segmentation results for Fig. 6.22(a) and Fig. 6.22(b) respectively. Figure 6.24(a) is the image of a woman. Fig. 6.24(b) is the enhanced image. Fig 6.24(c)-(d) shows the segmentation results for Fig. 6.24(a) and Fig. 6.24(b) respectively. Fig. 6.25(a) is the image of a stork bird in a field. Fig. 6.25(b) is the enhanced image. Fig 6.25(c)-(d) shows the segmentation results for Fig. 6.25(a) and Fig. 6.25(b) respectively.

The MSFCM algorithm described in section 5.2 and the NAFCM algorithm described in section 5.3 has been tested in LAB color space on a synthetic image, a grayscale image and some color test images, and their results are shown in Fig. 6.26-6.35.

A synthetic image shown in fig. 6.26(a) is used to show how the three classes of the image, having intensity values 0, 255 and 128, are affected while clustering the image using various clustering method such as FCM, sFCM, MSFCM, and NAFCM when the noise is increased from (-40,40) to (-90,90). The effect of increasing noise is shown in the Fig. 6.26- Fig. 6.28. From the images in Fig.6.26 - Fig.6.28, the percentage of misclassified pixels in each three clusters present in the synthetic image is calculated.

Fig. 6.29 shows the comparison of segmentation results of region based segmentation, edge based segmentation and FCM clustering based segmentation under a low noise varying between (-25,25) on a grayscale image(weld crack image).

The color images considered as test images are shown in Fig. 6.20- Fig.6.35. Fig. 6.30(a) shows an woman’s image with a uniform random noise with magnitude varying between (-35,35), in this image our region of interest is the skin color. Fig. 6.30(b)-6.30(e) shows the output of applying FCM, sFCM, MSFCM and NAFCM respectively on the input image. Fig. 6.30(f)-6.30(i) are the segmented images after applying FCM, sFCM, MSFCM and NAFCM algorithm respectively. The second test image in Fig.6.31(a) is the aerial view of a cross-road with a uniform random noise varying between (-35,35). The clustering result of applying FCM,sFCM, MSFCM and NAFCM is shown in Fig.

6.31(b)-6.31(e) respectively. Fig.6.31 (f)-(i) shows the segmented result of the four cases respectively. The test image in Fig.6.32 (a) is that of a human hand with a huge green color plastic ring. The image is corrupted with a noise of (-45,45). The green color plastic ring is our object of interest in this image. The segmentation result of all the algorithms has been shown in fig.6.32 (f)-6.32(i).The test image shown in Fig. 6.33 (a) is that of bacteria with a noise of (-90,90). The bacteria image is separated using all the four methods and the segmentation results are shown in Fig 6.31(f)-(i). The test image in Fig. 6.33(a) is image of a stork bird in a field with a noise of (-60,60).The image is first enhance using CLAHE algorithm as in Fig.6.25(b). The bird is separated from its background (field) using all the four clustering methods and the results are shown in Fig. 6.30(f)-(i).The test image in Fig. 6.35(a) is that of a woman. The skin color is clustered using the four algorithms under a random noise varying between (-90,90). The image is enhanced first using CLAHE algorithm as in Fig.6.24 (b) and then the four clustering algorithms are applied in Fig. 6.35(b)-6.35(e). The segmentation results are shown in Fig. 6.35(f)-6.35(i).

Fig. 6.36-Fig 6.39 shows an application of the NAFCM algorithm in detecting the tumor growth by observing tumor images showing different symptoms as discussed in section 5.2.4. Fig. 6.36(b) shows an example of asymmetrical shape of the mole with a noise of (-60,60) in which one half is different from the other may indicate melanoma. Here, the left side of the mole is dark and a little raised, whereas the right side is lighter in color and flat. Fig. 6.36(c)-(d) are the results of using FCM for clustering of the image. The clusters formed by using NAFCM algorithm is shown in Fig.6.36(g)- 6.36(i). Fig. 6.37(b) is an example of a growth with irregular border having a noise of (-60,60). The clusters formed by using NAFCM algorithm is shown in Fig.6.36(f)- 6.36(g). Fig. 6.38(b) is an example of changes in colour of a mole with a noise of (-90,90).Fig.6.38(c)-6.38(l) shows the various clusters and segmented image using FCM and NAFCM respectively. Fig.6.39(b) is a case where we take the diameter of the mole into consideration to know whether it can develop into skin cancer. Here the image has a noise of (-90,90). Fig.6.39(c)-6.39(h) shows the various clusters and segmented image using FCM and NAFCM respectively.

Table 6.3 shows the quantitative analysis of all the algorithms. A high Vpc and a low Vpe gives good clustering result. Table 6.4 shows the percentage of misclassified pixels in each cluster for increasing noise condition.

Fig. 6.39 and Fig 6.40 shows the result of applying morphological operators for segmentation of tumor images as described in section 5.2.3. Table 6.5 shows that NAFCM gives good clusters (high V_{pc} and low V_{pe}) while using it for segmentation of melanoma images.

6.2 Discussion

Two examples for region growing have been shown in Fig 6.1 and Fig 6.2. In the first case, Fig. 6.1, the seed point under consideration is single pixel intensity. In the second case, Fig. 6.2, an array of seed points has been considered, where pixels are added to a region if any of the pixels in its four neighbourhood satisfies a predefined condition. The first test image is an infrared image of an aluminium weld with porosity or crack. A threshold value of 65 and pixel intensity of 255(brightest pixels signify crack) are taken as condition for region growing. The second test image is a scenery image. The region is iteratively grown by comparing all unallocated neighbouring pixels to the region.The distance between a pixel's intensity value and the region's mean,is used as a measure of similarity. The pixel with the smallest distance measured this way is allocated to the respective region.This process stops when the intensity difference between region mean and new pixel become larger than a certain threshold.Region maximum distance is taken as 0.3. Figure 6.3 shows how the weld crack is segmented by using region split and merge with a standard deviation greater than 10 and mean intensity greater than 15. In Fig. 6.4 (b) it is observed that we are able to detect the isolated star using point detector mask. In Fig. 6.5(b)-(c) it is observed that the horizontal lines in the image are easily recovered. By observing Fig 6.6(b)-(f) it can be said that, Canny operator performs the best in detecting all edges, but the problem with it is that it gives false edges also. The Canny edge detector also requires the set of two threshold every time by the user.

The FIS based edge detection described in section 3.1 is tested with different images, its performance being compared to that of the other derivative based edge detectors like, Sobel operator and Canny edge detector. It is observed in Fig. 6.7 that the Sobel operator does not allow edges to be detected in the region where the transition from high gray level values of image pixels to low gray level values of image pixels is blurred. The Canny edge detector is able to detect all edges but it also gives some false edges along with the true edges. The FIS system in turn, allows edges to be almost detected even in the low contrast regions without the output image being much affected by noise. But still it is unable to detect true edge completely. In case of Fig. 6.8 we observe that again the Sobel operator is not able to detect edges in the low contrast region. The Canny edge detctor is able to detect some edge pixels in the low contrast regions of the image but it is unable to detect any eges in high contrast region. The FIS system is able to detect edge pixels in the low contrast region and some pixels even in the high contrast regions.In Fig. 6.9 it is seen that the original image is quite blurred in nature. The Sobel edge detector is again unable to detect edges in the regions where the image starts getting more blurred. The Canny edge detector is the best performer here as it detects edges even in the blurred region, even though it has a disadvantage of detecting false edges.

The FMFED algorithm described above has been tested on some test images and its qualitative performance is compared to two popular edge detectors – Sobel and Canny edge detectors[57]. By the visual comparison of all the algorithms results Fig. 6.10- 6.14 (b-d), it is observed that the sobel edge detector operator performs the worst among all as it is unable to detect true edge pixels in certain areas. The Canny edge detector on the other hand is able to detect all the edge pixels but the problem with this method is that it detects false edges too. These false edges give wrong information about the original objects approximate shape. False edges are also a liability in cases where edge detection is used for image compression. In case of all test images considered here it is seen that we get too many edge pixels in the cases where Canny edge detector is applied to the original image. Whereas, when Sobel operator is applied to the same images, certain important edge information has been lost. The FMFED algorithm is a good. By observing the result of applying morphological operators on the edge detcted images [Fig. 6.15(a)-6.15(c)] shown in Fig. 6.15(d)-6.15(f), it can be said that morphological operators are not the best way to fill the gaps in the edge images and hence there is scope for applying some other edge linking algorithm.

The FCM based segmentation described in chapter 4, section 4.4 has been tested on some colour test images in LAB color space and their results are discussed here. Apart from the reason that CIELAB color space provides perceptually uniform space, it is also observed that using RGB color space the clusters that are formed are not correct as shown in Fig. 6.16 and Fig. 6.17, here comparison is made between segmentation in RGB color space and LAB color space. The same results are proved by observing quantitative comparison between the two color space is done in Table 6.1 and Table 6.2 as the LAB color space shows high V_{pc} and low V_{pe} value (condition for good clustering).

The images considered as test images for applying FCM algorithm are shown in Fig.6.18 (a) – Fig.6.21 (a). Fig.6.18(a) is an image of some vegetation in desert area. Fig. 6.18(b)-(d) by using just c=2, the sand and the green vegetation has been segmented satisfactorily. Figure 6.19(a) is the image of a women, here we want to segment the skin color, which can be done easily by choosing selecting c = 3. Fig. 6.20(a) is a biomedical image showing hyper pigmentation of skin of an old lady. The cheek region shows dark patches due to hyperpigmentation. By selecting c=2, the affected skin which is brown or dark in color is separated from the pink skin which is unaffected by hyperpigmentation. Fig. 6.21(a) is an image containing a bird flying in sky. Selecting c = 2 we are able to separate the bird from the sky.

The CLAHE based FCM algorithm described in section 4.6 has been testedin LAB color space on some color test images and their results are discussed. The CLAHE algorithm is applied only to the luminosity layer L. This is because the enhancement of the image depends on the brightness level of the image pixels. The images considered as test images here are shown in Fig 6.22-6.25(a). Fig. 6.22(a) is a close aerial view of a landscape, which has a water body along with some dry areas. Taking c=3, segments the water area (dark blue color in the segmented image) from the dry area. Figure 6.23(a) is the image of a person’s cheek region suffering from hyper-pigmentation. The extent of affected skin (shown by dark blue color in the segmented image) is known more accurately in the case when segmentation is done after image enhancement. Figure 6.24(a) is the image of a woman. Taking the number of clusters c=3, the skin is correctly classified and segmented (yellow color in segmented image) after enhancement of the image. Fig. 6.25(a) is the image of a stork bird in a field. Taking the number of clusters c=3, the stork bird is almost correctly classified and segmented after enhancement of the image [sky blue color Fig. 6.25(d)].

The MSFCM algorithm described in section 5.2 and the NAFCM algorithm described in section 5.3 have been tested on a synthetic image, a greyscale image, and some color test images in LAB color space, and their results are discussed. In Fig. 6.26- Fig. 6.28 it is observed that under low noise condition the best clusters are formed for MAFCM algorithms and NAFCM gives the best clusters under high noise condition. From the images in Fig.6.26 - Fig.6.28, the percentage of misclassified pixels in each three clusters present in the synthetic image is calculated.

In Fig. 6.29 (weld crack image) it is observed that the proposed method MSFCM performs the best under low noise condition. The region growing based segmentation method completely fails to recognise regions under noisy conditions. The split and merge based segmentation too fails to identify the weld crack correctly.

The color images considered as test images are shown in Fig. 6.30- Fig.6.35. In Fig. 6.30 it is observed that the edges of the clustered output are better preserved in case of the MSFCM algorithm as compared to the case when FCM, sFCM and NAFCM algorithms have been applied for a noise of (-35,35). In Fig.6.31 it is observed that the clustering result obtained by applying FCM gives the worst result as it is not able to reduce the noise present in the image during clustering operation for a noise of (-35,35). In case of Fig.6.31(d) we observe that the edges of inner circle and inner triangle of the road is smoother as compared to Fig.6.31(b), Fig. 6.31(c) and Fig. 6.31(e) where FCM, sFCM and NAFCM algorithms are applied. The test image in Fig.6.32 (a) is that of a human hand with a huge green color plastic ring. The image is corrupted with a noise of (-45,45). The green color plastic ring is our object of interest in this image. The MSFCM algorithm is able to cluster the object of interest accurately and the edges are also preserved well as compared to the other algorithms. Fig. 6.33 (a) is that of bacteria with a noise of (-90,90). The bacteria is separated from its background using all the three methods. It is observed that the NAFCM algorithm is able to retain the boundary of bacteria more effectively as compared to other methods.

Fig. 6.34(a) is image of a stork bird in a field with a noise of (-60,60).The image is first enhanced using CLAHE algorithm as in Fig.6.25(b). The bird is separated from its background (field) using all the four clustering methods. It is observed the edges are well preserved using NAFCM algorithm.

Fig. 6.35(a) is that of a woman. The skin color is clustered using the four algorithms under a random noise varying between (-90,90). The image is enhanced first using CLAHE algorithm as in Fig.6.24(b) and then the four clustering algorithms are applied. The NAFCM algorithm is seen to perform the best clustering to segment the skin of the woman.

Fig. 6.36-Fig 6.39 shows an application of the NAFCM algorithm in detecting the tumor growth by observing tumor images showing different symptoms as discussed in section 5.2.4. Fig. 6.36(b) shows an example of asymmetrical shape of the mole with a noise of (-60,60) in which one half is different from the other may indicate melanoma. Here, the left side of the mole is dark and a little raised, whereas the right side is lighter in color and flat. Fig. 6.36(c)-(d) are the results of using FCM for clustering of the image. By observing these images we cannot say anything about the irregularity of the growths shape. But using the NAFCM algorithm we observe in Fig.6.36(g) and Fig. 6.36(h) that the two halves of the growth are very different from each other hence it has chances of developing into melanoma. Fig. 6.37(b) is an example of a growth with irregular border having a noise of (-60,60). The irregular border can be very easily determined using proposed method as shown in Fig. 6.37(h). Fig. 6.38(b) is an example of changes in colour of a mole with a noise of (-90,90). The different colours present in the growth are not easily observed using naked eyes or standard FCM method of clustering, whereas by using the NAFCM method we observe more than two color or uneven distribution of color present in the affected area. Fig.6.39(b) is a case where we take the diameter of the mole into consideration to know whether it can develop into skin cancer. Here the image has a noise of (-90,90). For measuring the diameter of the growth the border of the growth has to be known accurately but because of noise it is impossible to know the

borders clearly. NAFCM algorithm is able to find almost accurate border even under high noise as shown in Fig. 6.39(h). Table 6.5 shows that NAFCM algorithm gives good clustering results (high V_{pc} and low V_{pe}) for high noise condition in melanoma image segmentation

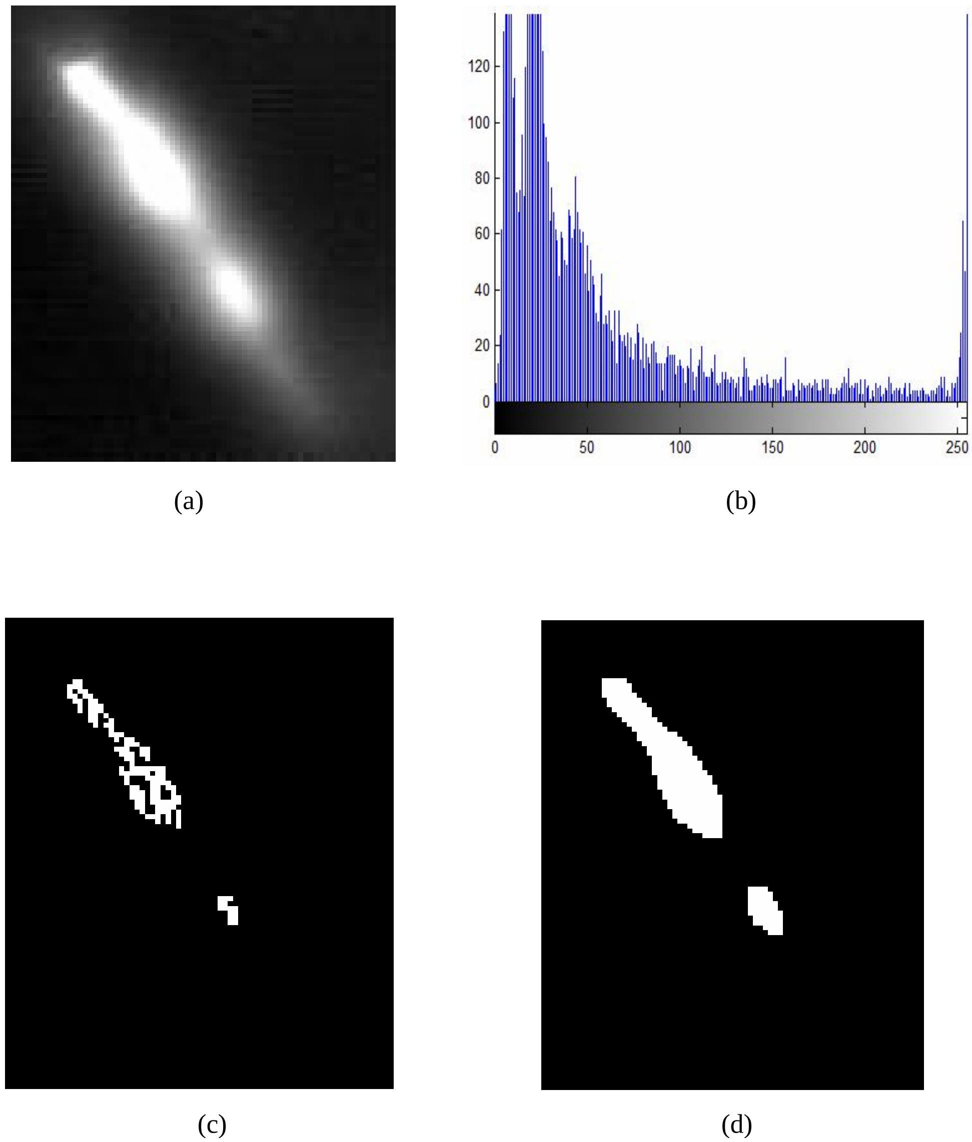
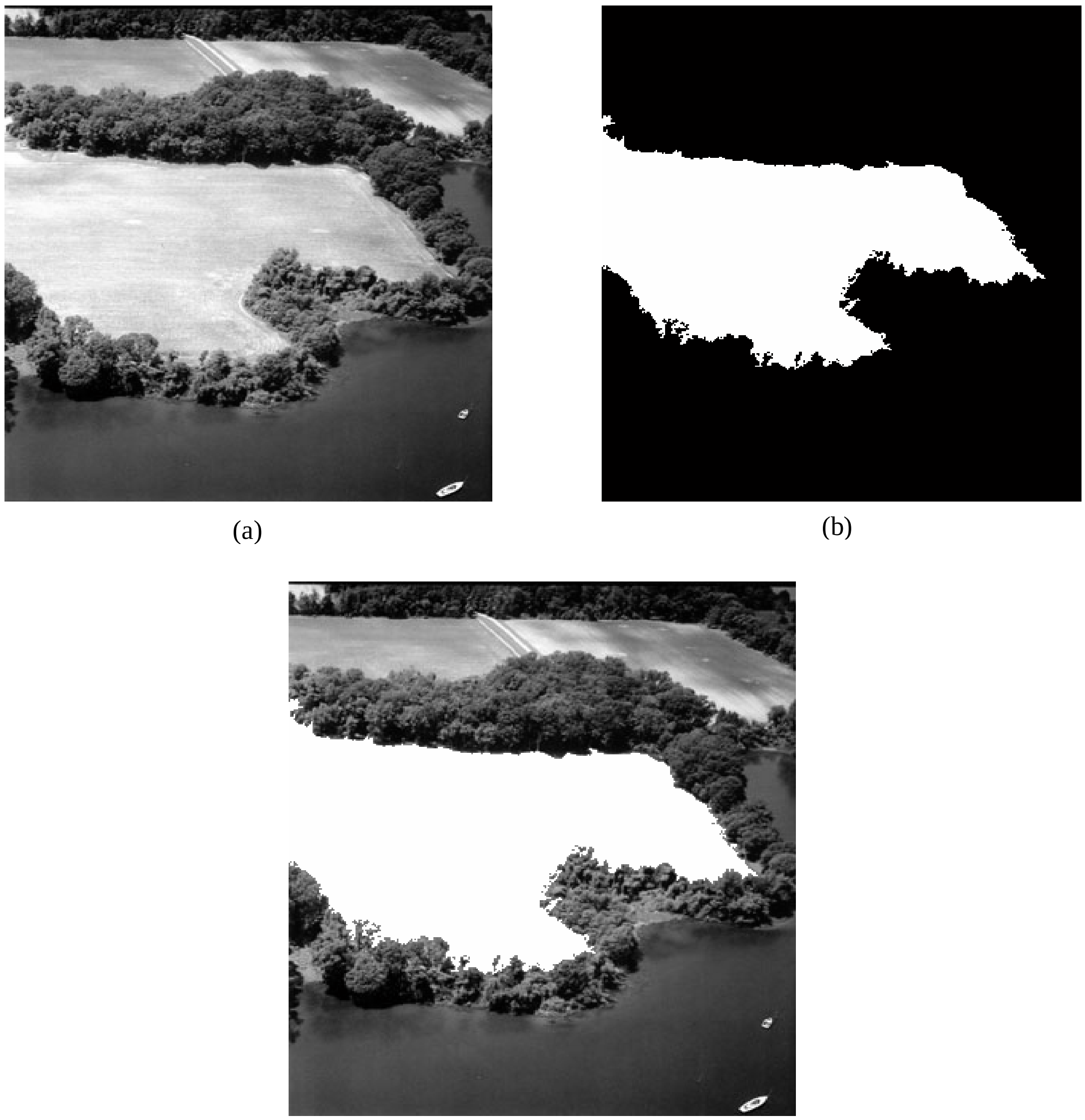
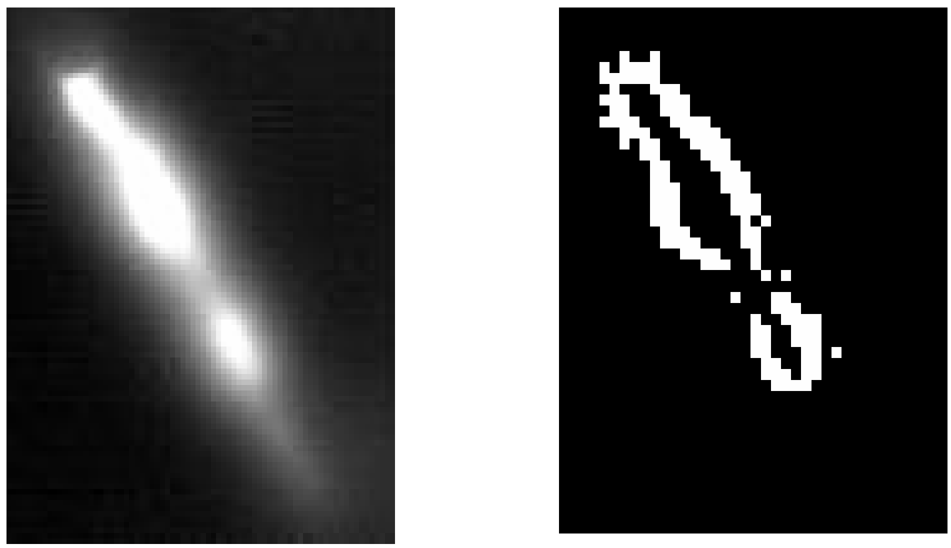


Figure 6.1 (a) Original infrared image of an aluminium metal casting with porosity (b) Histogram of figure 2.1(a). (c) Seed points. (d) Result of region growing



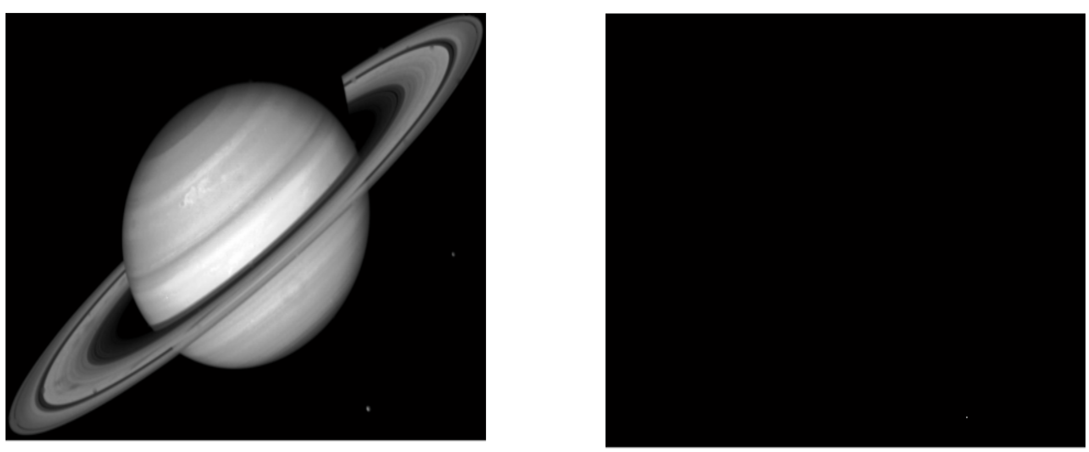
(a) (b) (c)

Figure 6.2 (a) Original scenery image (b) Array of seed points (c) Result of region growing



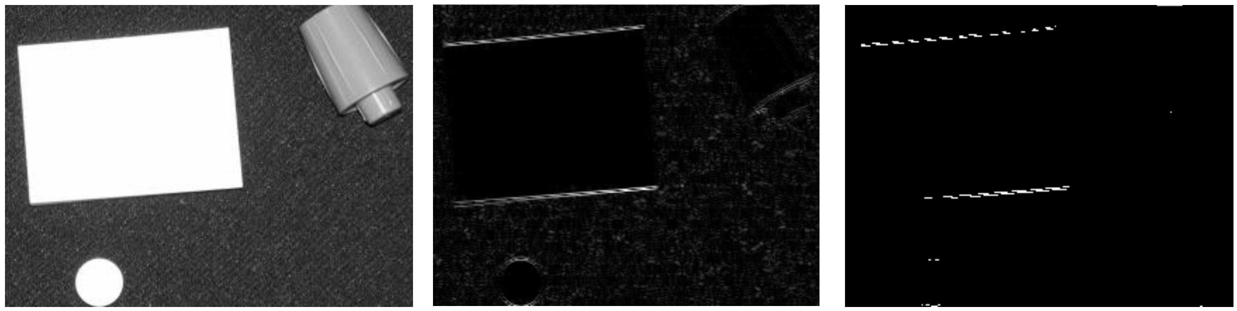
(a) (b)

Figure 6.3 (a) Image of crack in a weld (b) Result of region split and merge with a standard deviation >10 and mean intensity >15.



(a) (b)

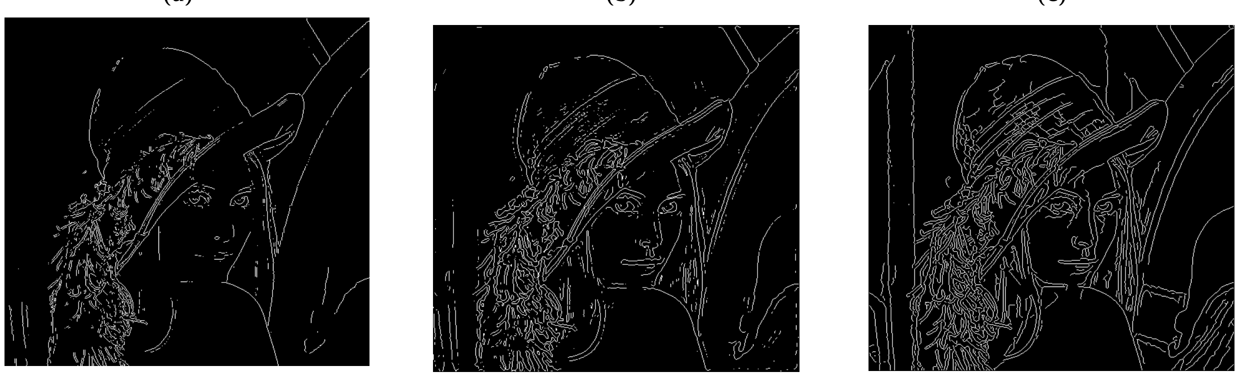
Figure 6.4 (a) Image of planet Saturn with a small isolated white star on the bottom right side. (b) Result of point detection



(a) (b) (c)

Figure 6.5. (a) Image of pill set (b) Result of running a horizontal line detection mask through the image (c) Result of thresholding fig. (b) with mean of maximum and minimum value of pixels

(a) (b) (c)



(d) (e) (f)

Figure 6.6 (a) Original lena image (b) Output of Roberts edge detector (c) Output of Prewitt edge detector (d) Output of Sobel detector (e) Output of Log detector (f) Output of Canny edge detector

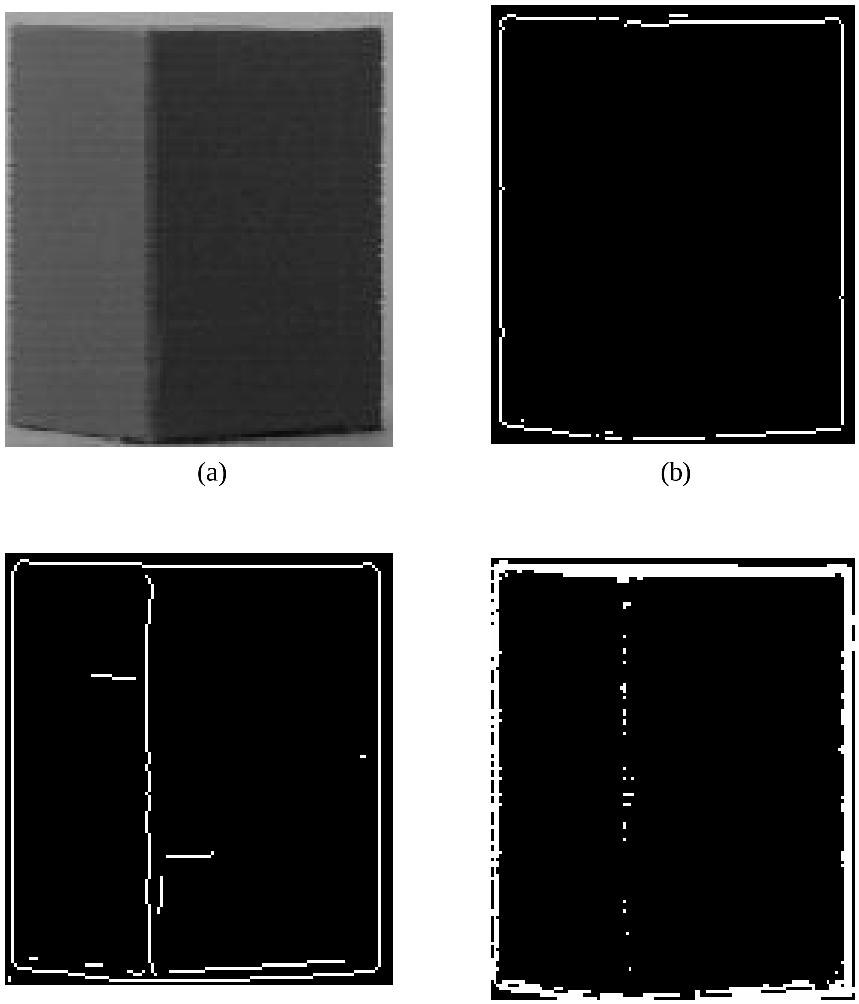
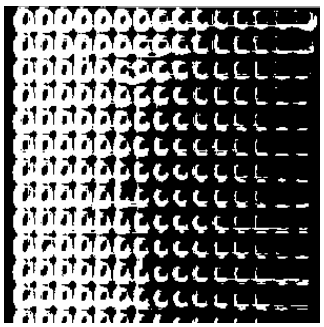


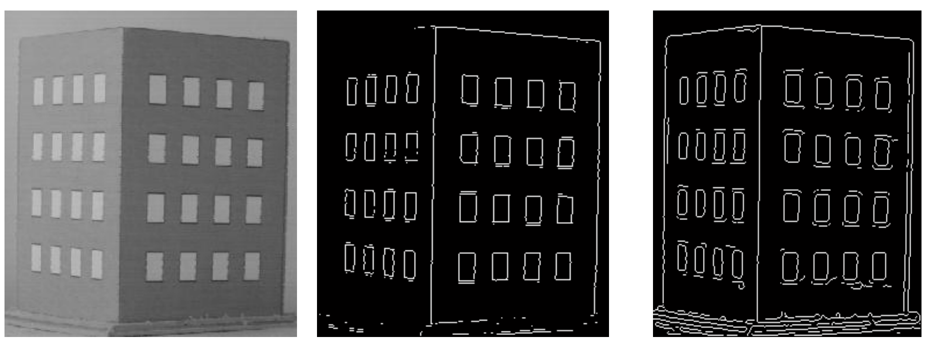
Figure 6.7: (a) A wooden block's image. (b) Edges detected by the Sobel operator. (c) Edges detected by the Canny edge detector (d) Edges detected by the studied FIS system.

(a) (b) (c)

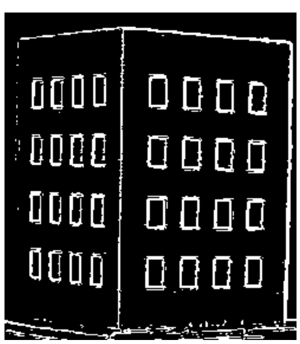


(e)

Figure 6.8: (a) A digital cameras calibration pattern's image. (b) Edges detected by the Sobel operator . (c) Edges detected by Canny edge detector. (d) Edges detected by the studied FIS system.

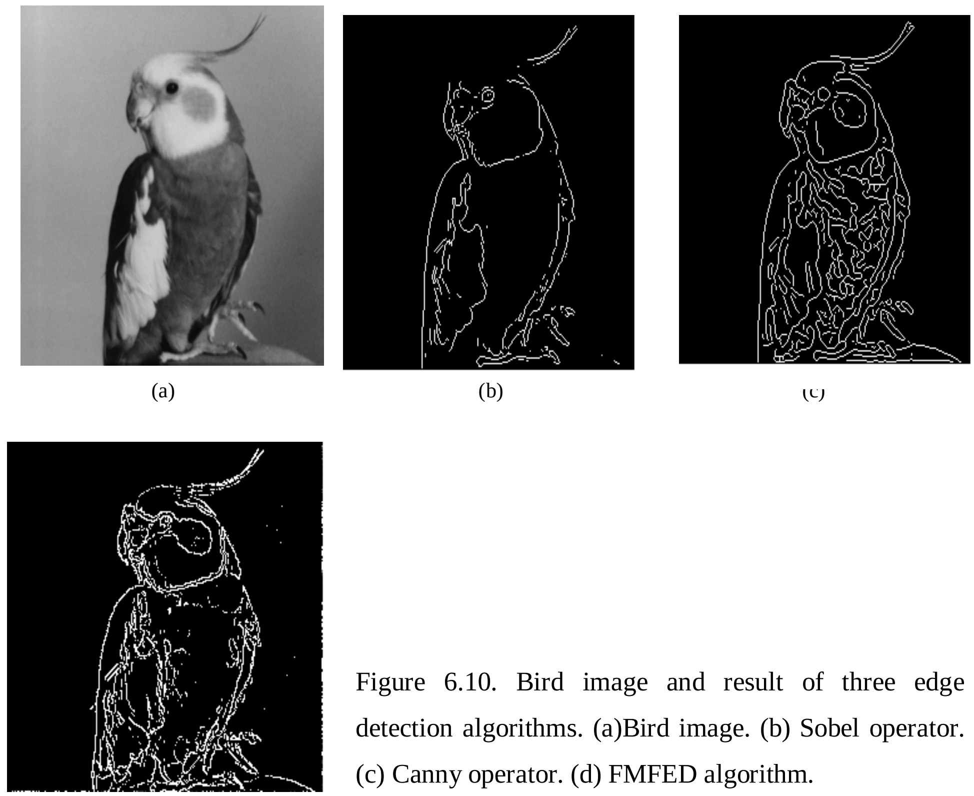


(a) (b) (c)



(d)

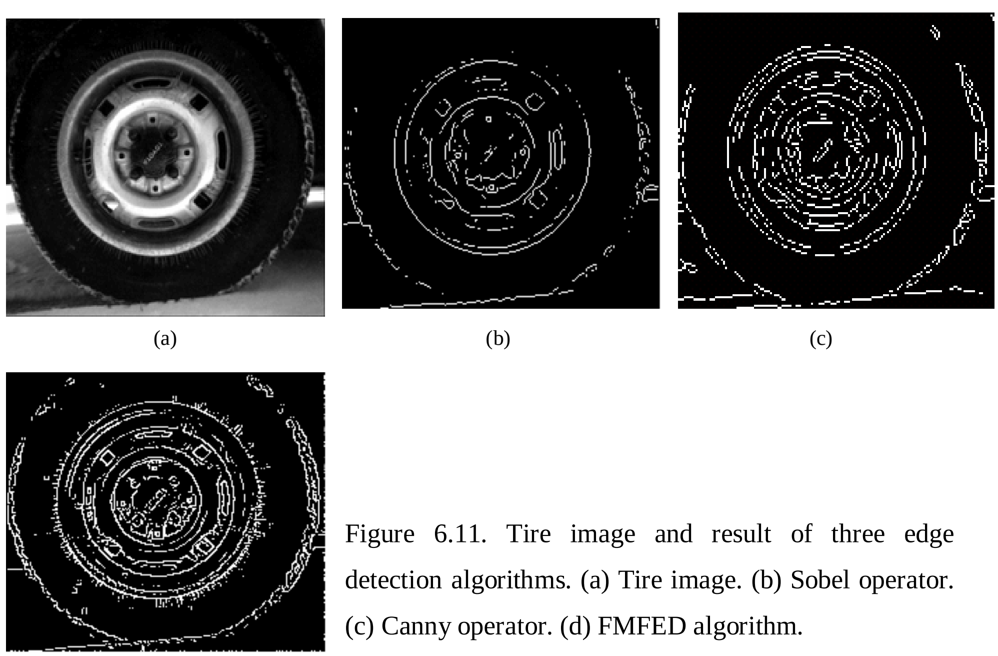
Figure 6.9 (a) A digital cameras calibration pattern's image. (b) Edges detected by the Sobel operator. (c) Edges detected by the Laplacian of Gaussian operator. (d) Edges detected by the studied FIS



(a) (b) (c)

(d)

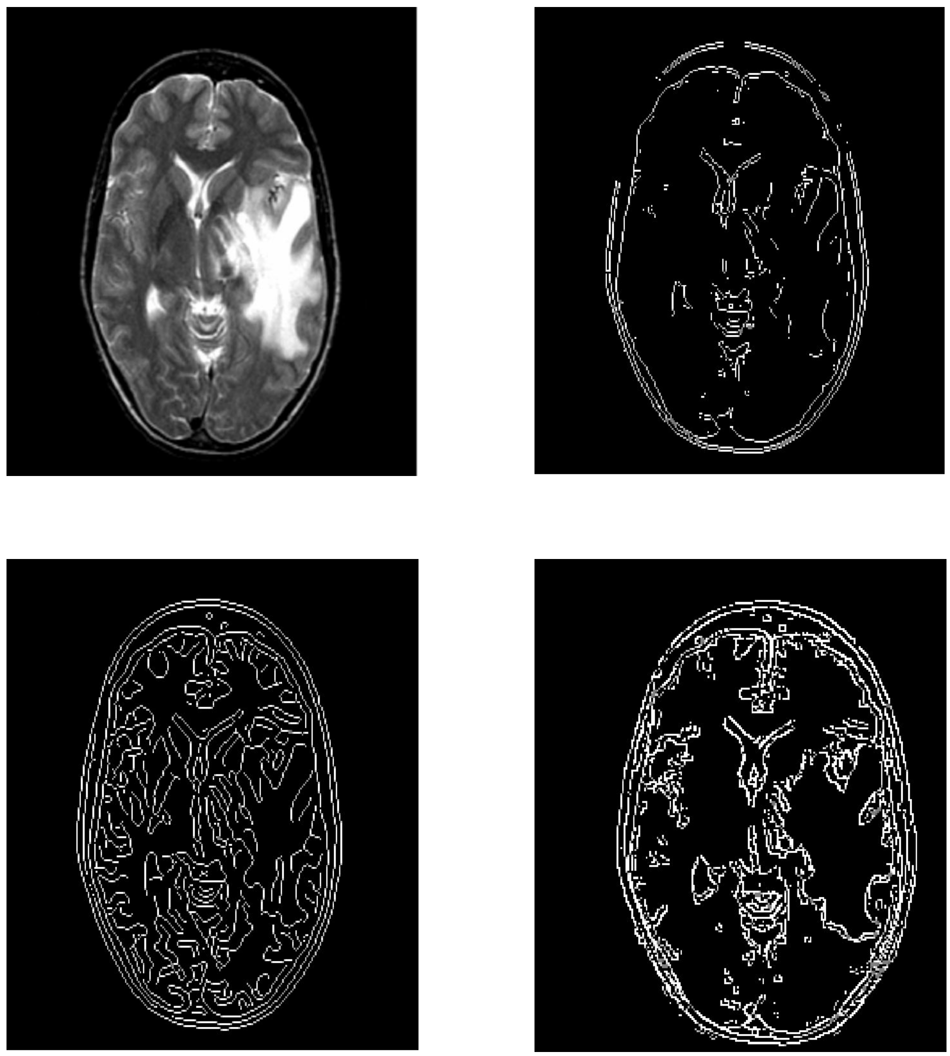
Figure 6.10. Bird image and result of three edge detection algorithms. (a)Bird image. (b) Sobel operator. (c) Canny operator. (d) FMFED algorithm.



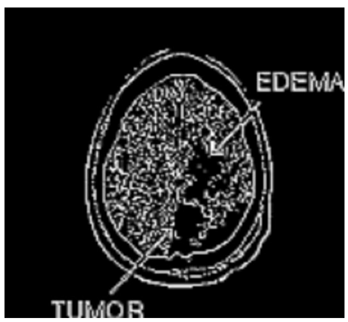
(a) (b) (c)

(d)

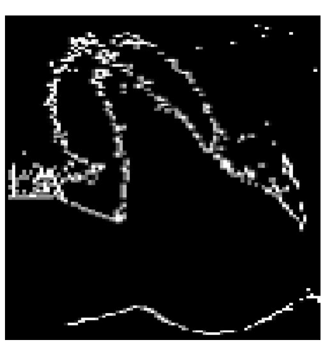
Figure 6.11. Tire image and result of three edge detection algorithms. (a) Tire image. (b) Sobel operator. (c) Canny operator. (d) FMFED algorithm.



a b
c d
Figure 6.12. MRI brain image and results of three edge detection algorithms. (a) MRI brain image. (b) Sobel operator . (c) Canny operator. (d) Proposed algorithm



a b
c d
Figure 6.13. X-ray image of brain and results of three edge detection algorithms. (a) X-ray image of brain. (b) Sobel operator. (c) Canny operator. (d) FMFED algorithm.



a b
b d
Figure 6.14. Dental X-ray image and results of three edge detection algorithms. (a) Dental X-ray image with abscess. (b) Sobel operator. (c) Canny operator. (d) FMFED algorithm.

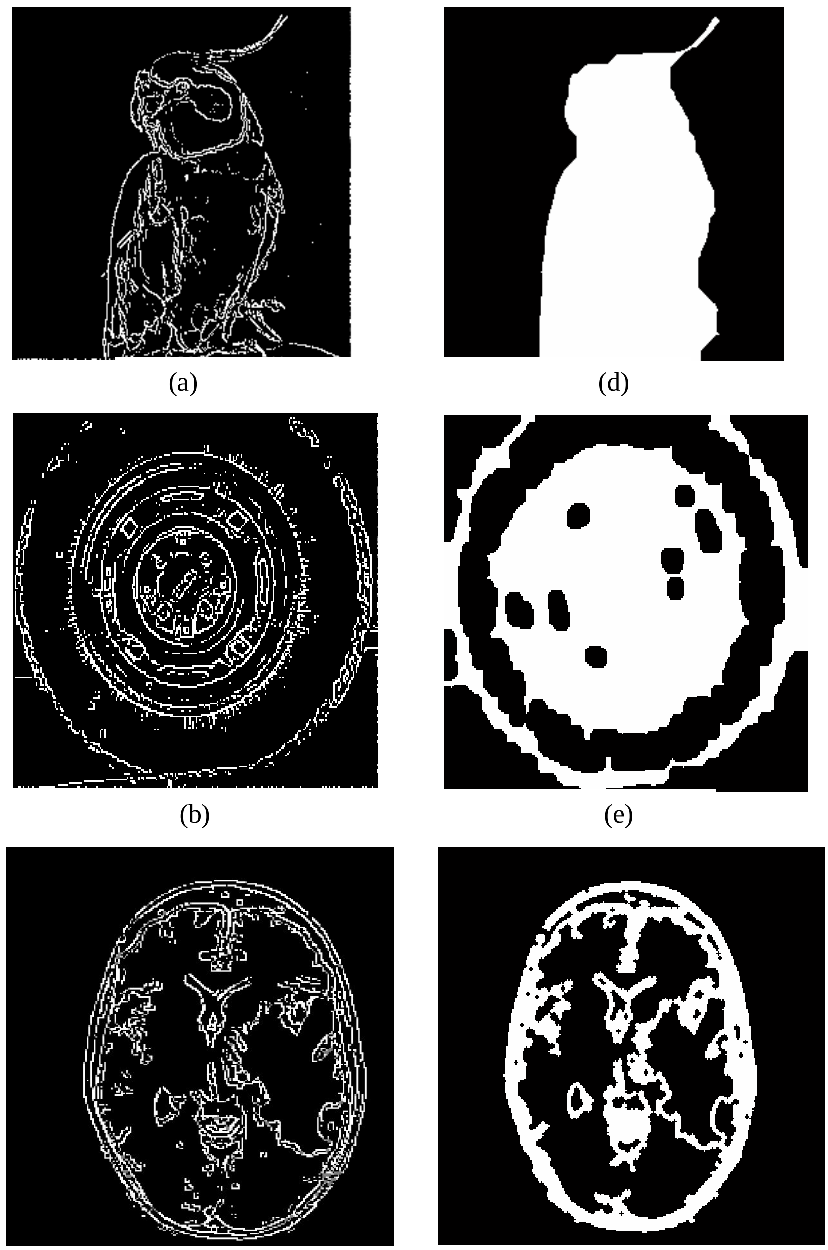


Figure 6.15 (a)-(c) Edges detected by FMFED algorithm (d)-(f) result of applying morphological operators for segmentation

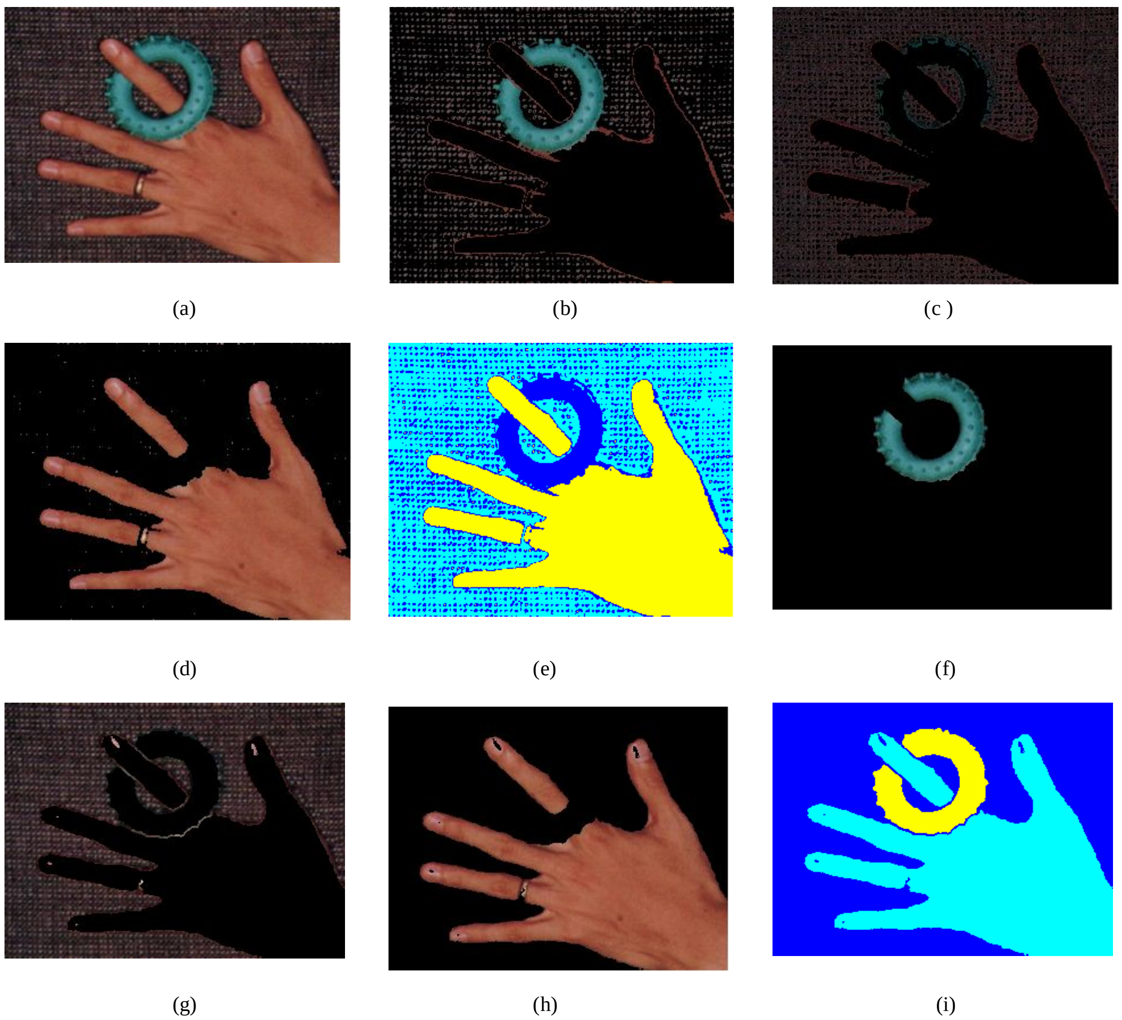


Figure 6.16 (a) Original image. (b)-(d) The three clusters for c=3 in case of RGB color space. (e) Segmented image in RGB color space. (f)-(h) The three clusters for c=3 in case of L*a*b* color space. (i) Segmented image in L*a*b* color space.

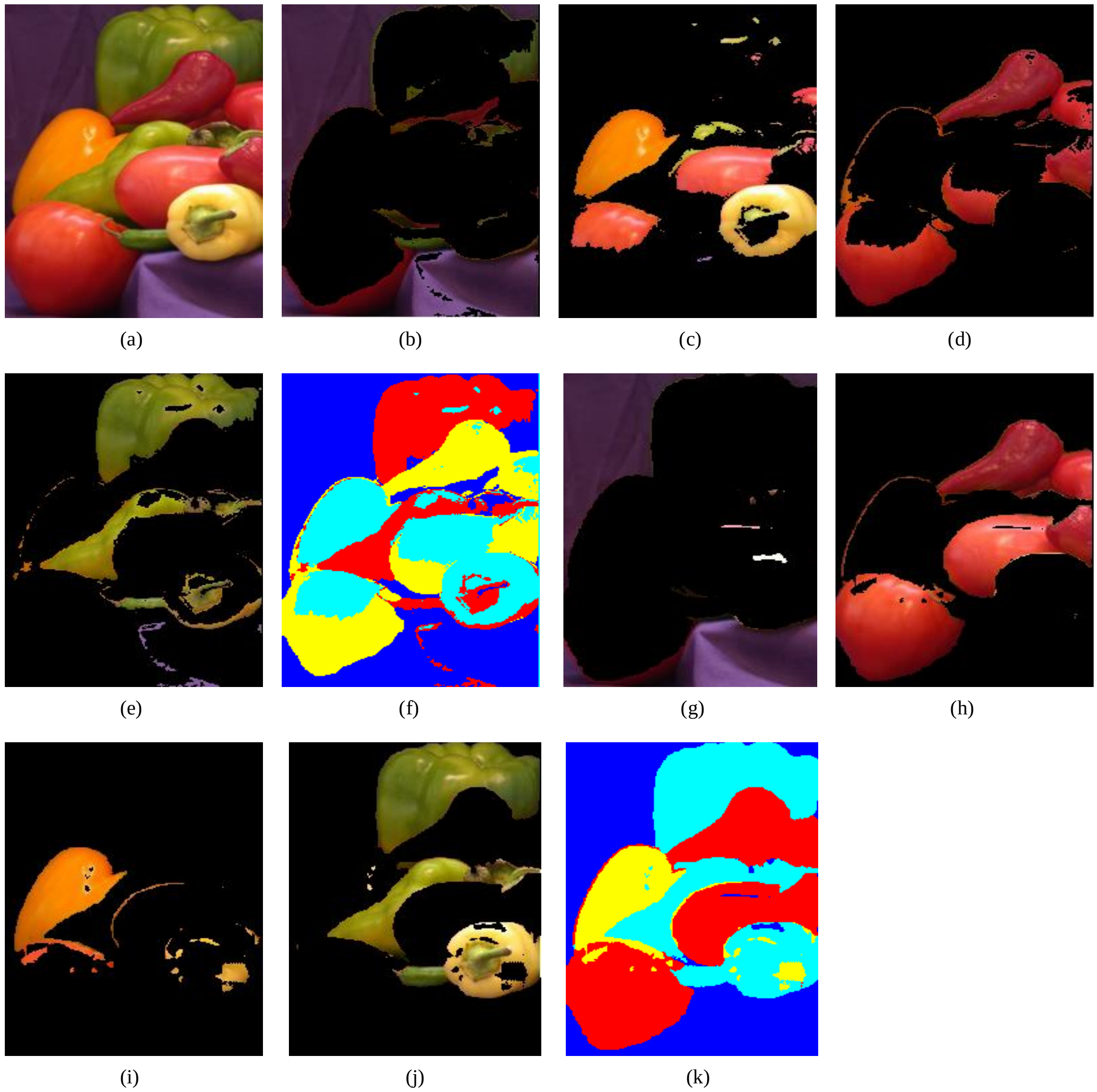
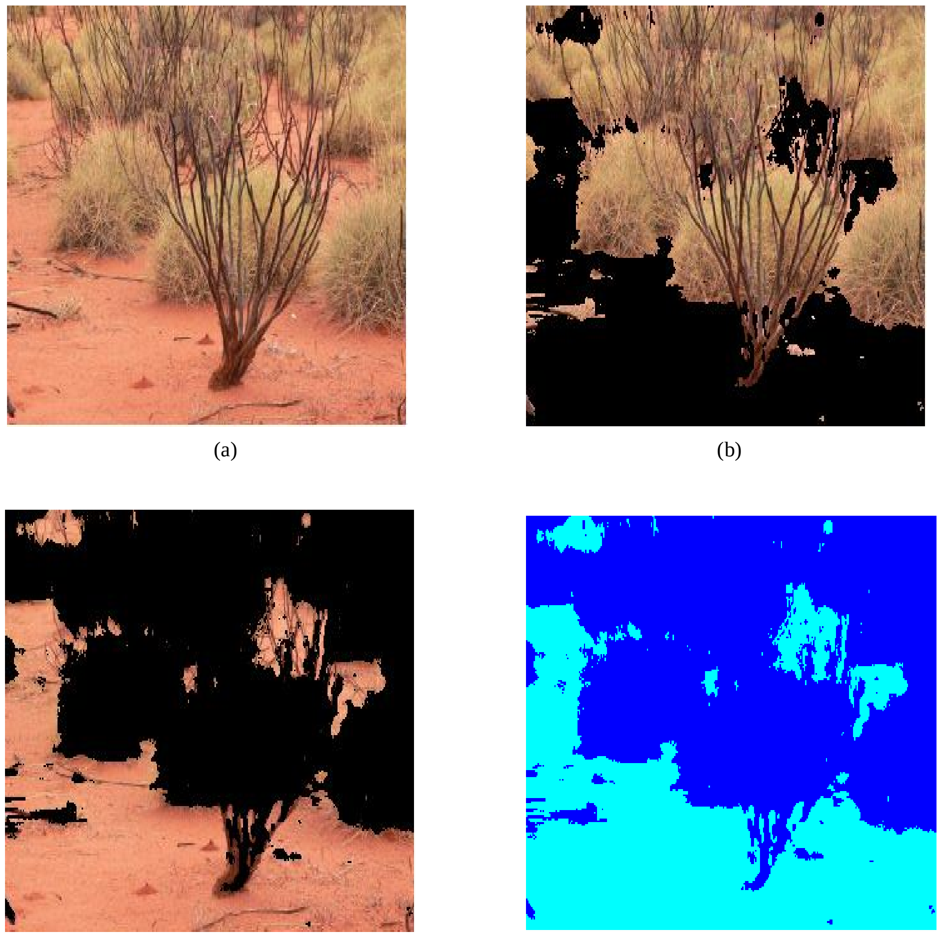


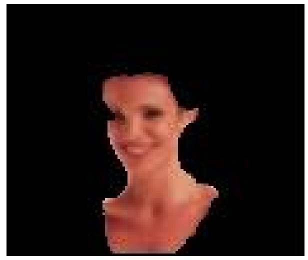
Figure 6.17 (a) Original image. (b)-(e) The four clusters for c=4 in case of RGB color space. (f) Segmented image in RGB color space. (g)-(j) The four clusters for c=4 in case of L*a*b* color space. (k) Segmented image in L*a*b* color space.



(a) (b) (c) (d)

Figure 6.18 (a) Original image. (b)-(c) The two clusters for c=2 (h) Segmented image


(a)


(b)


(c)


(d)


(e)

Figure 6.19 (a) Original image. (b)-(d) The three clusters for c=3 (e) Segmented image


(a)


(b)


(c)

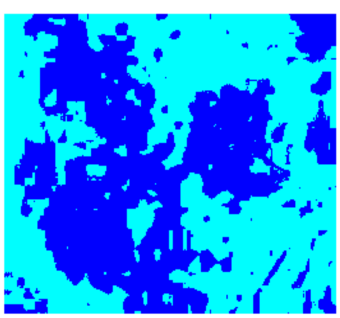

(d)

Figure 6.20 (a) Original image. (b)-(c) The two clusters for c=2 (e) Segmented image

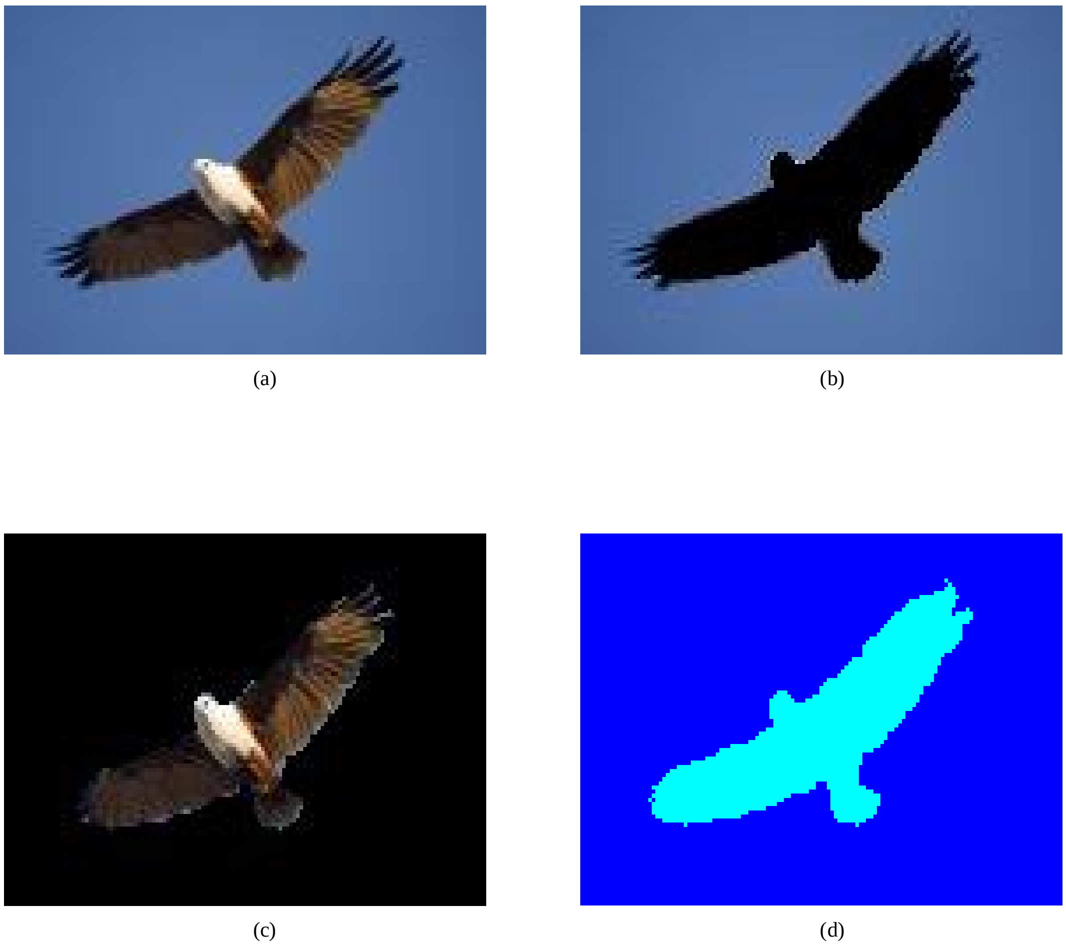


Figure 6.21 (a) Original image. (b)-(c) The two clusters for c=2 (d) Segmented image

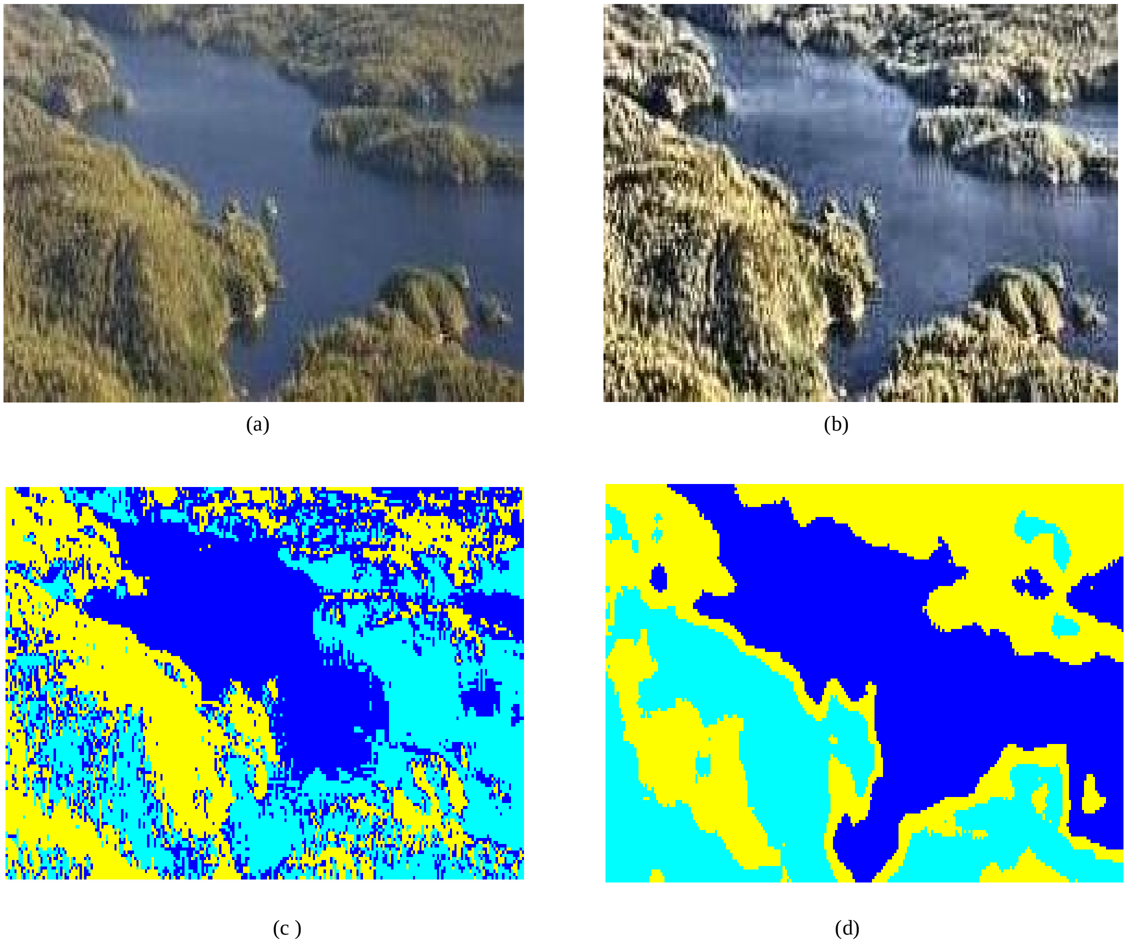


Figure 6.22. (a) Original image (b) Enhanced image (c) Segmented image without enhancement (d) Segmented image after enhancement

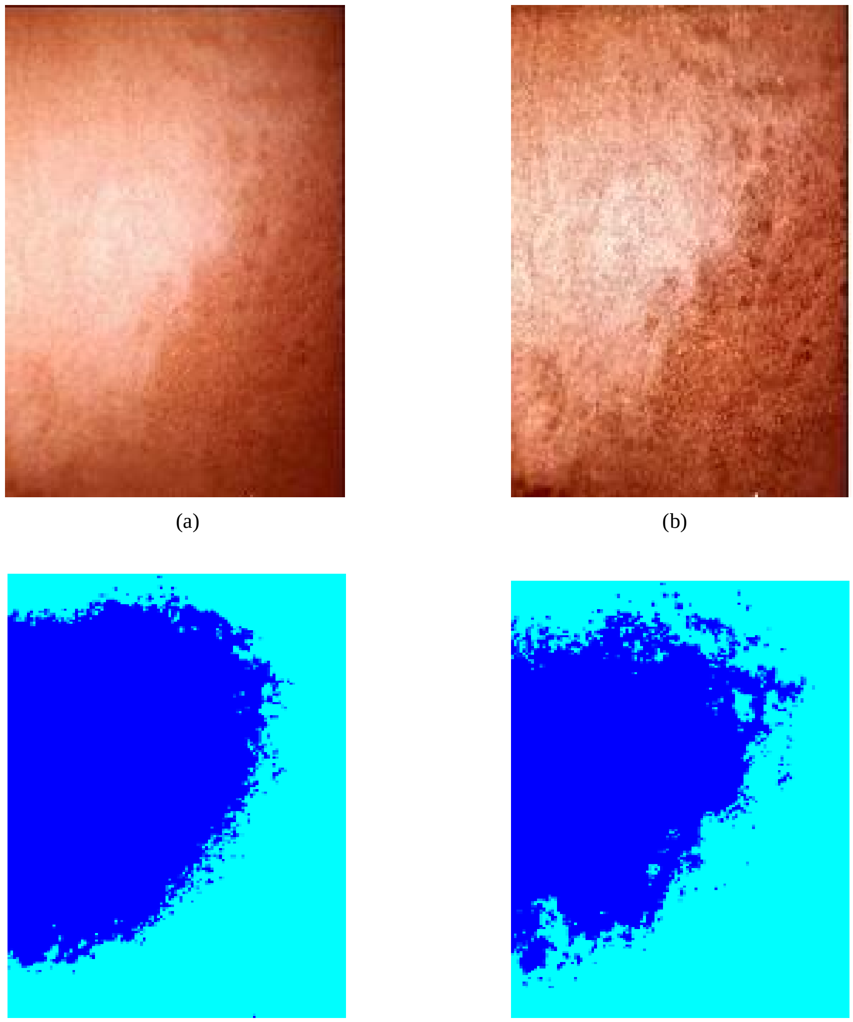
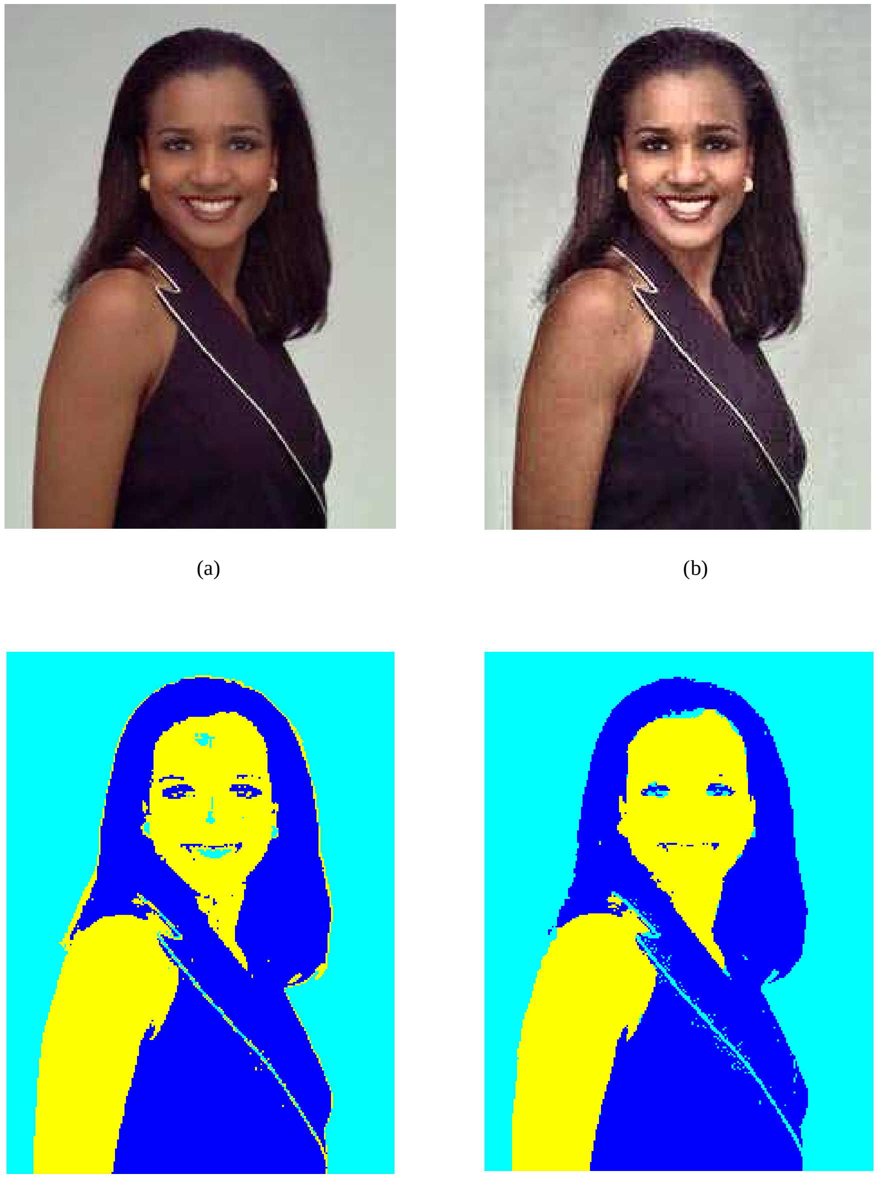


Figure 6.23. (a) Original image (b) Enhanced image (c) Segmented image without enhancement (d) Segmented image after enhancement



(a) (b)

(c) (d)

Figure 6.24. (a) Original image (b) Enhanced image (c) Segmented image without enhancement (d) Segmented image after enhancement

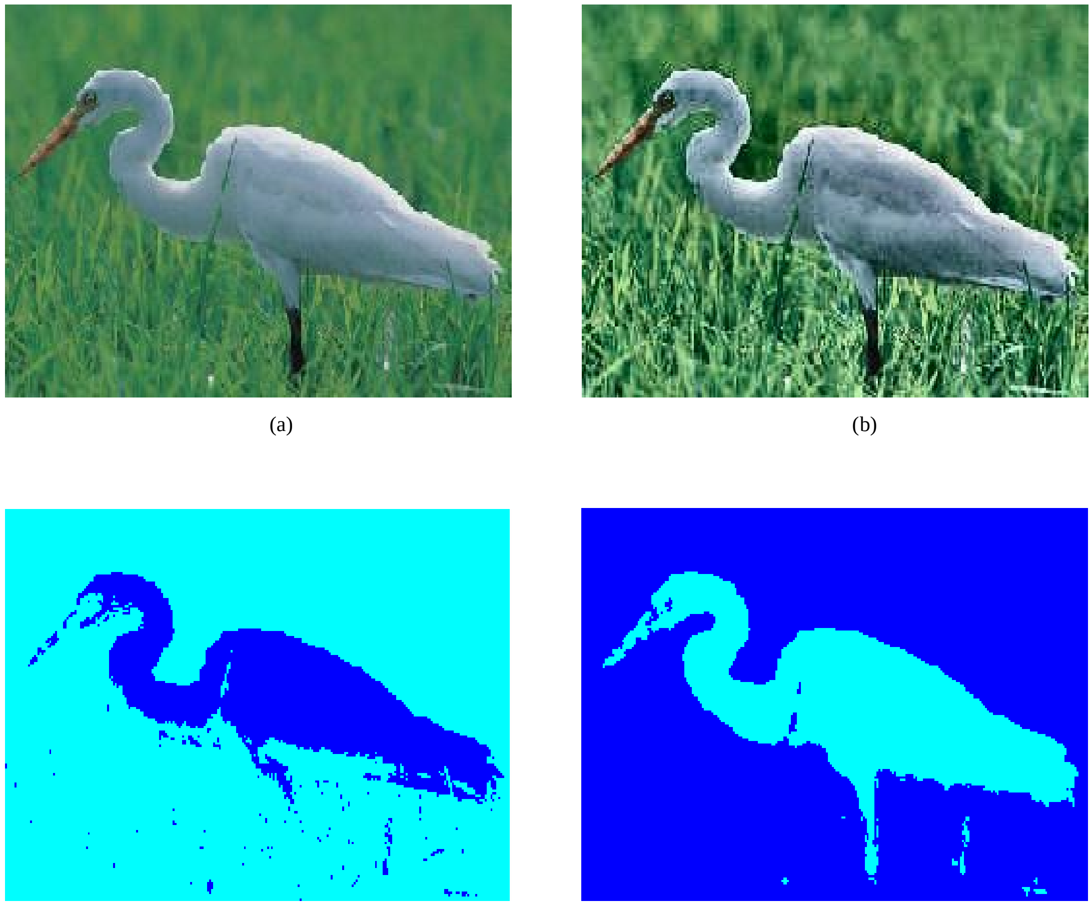
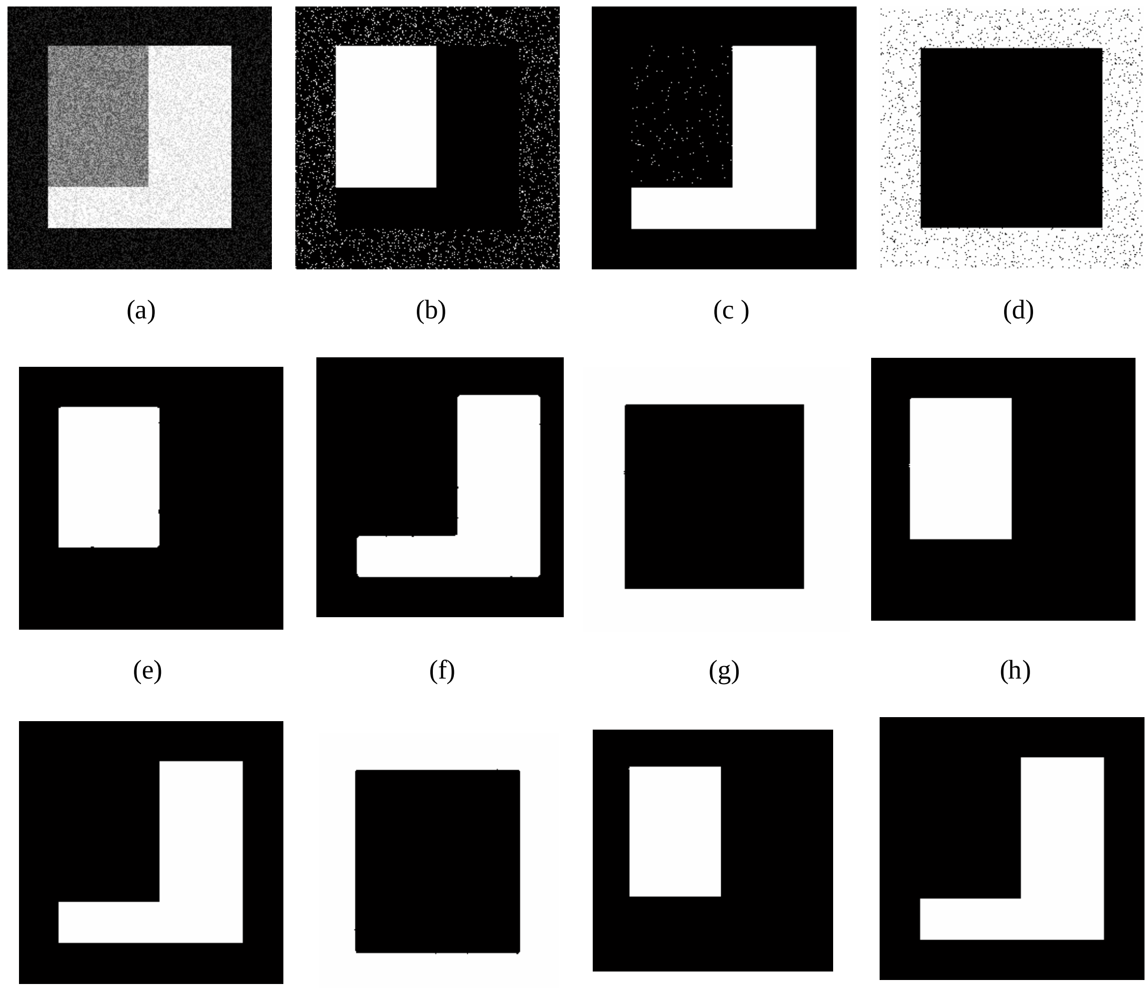
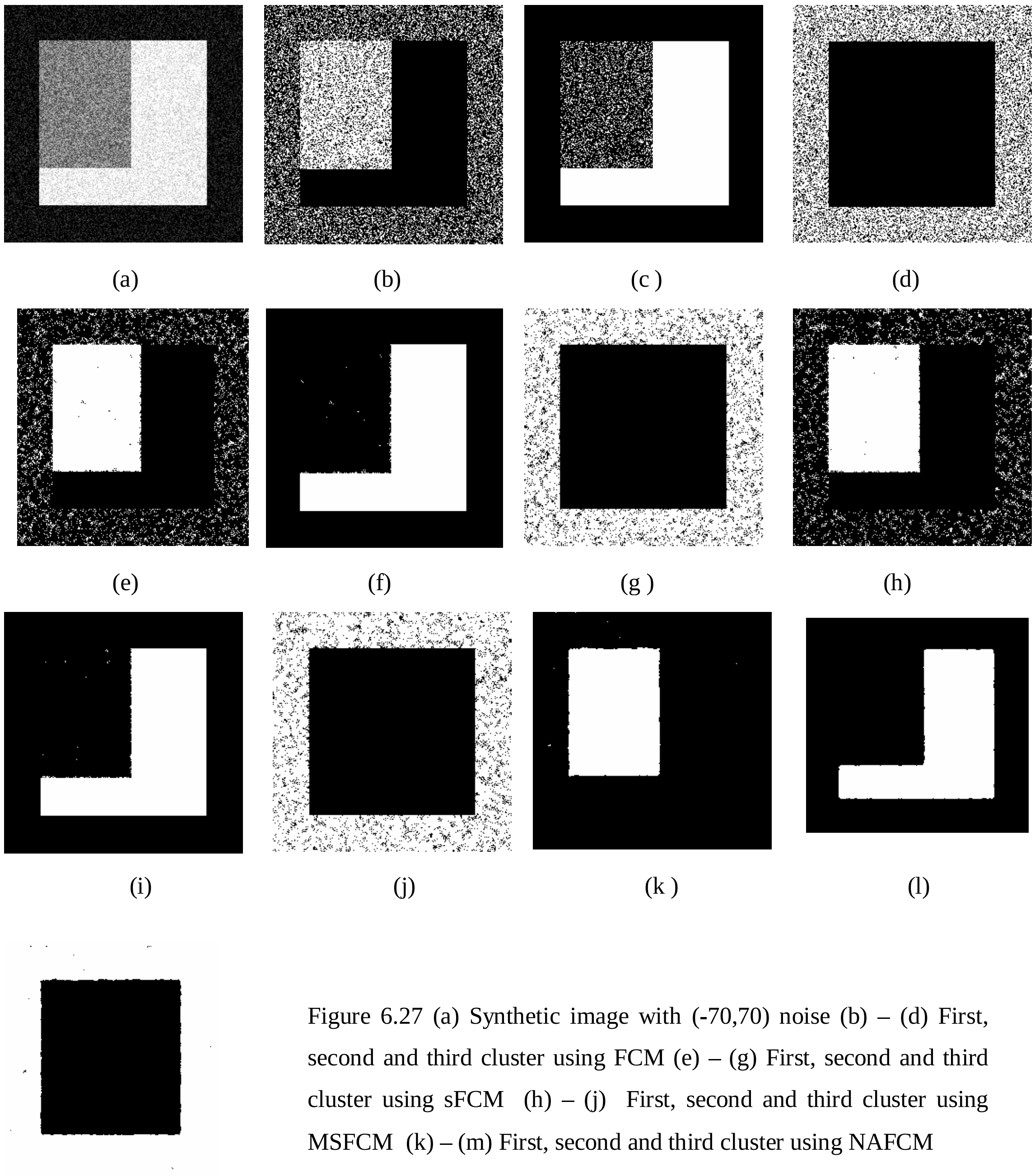


Figure 6.25. (a) Original image (b) Enhanced image (c) Segmented image without enhancement (d) Segmented image after enhancement



(m)

Figure 6.26 (a) Synthetic image with (-60,60) noise (b) – (d) First, second and third cluster using FCM (e) – (g) First, second and third cluster using sFCM (h) – (j) First, second and third cluster using MSFCM (k) – (m) First, second and third cluster using NAFCM



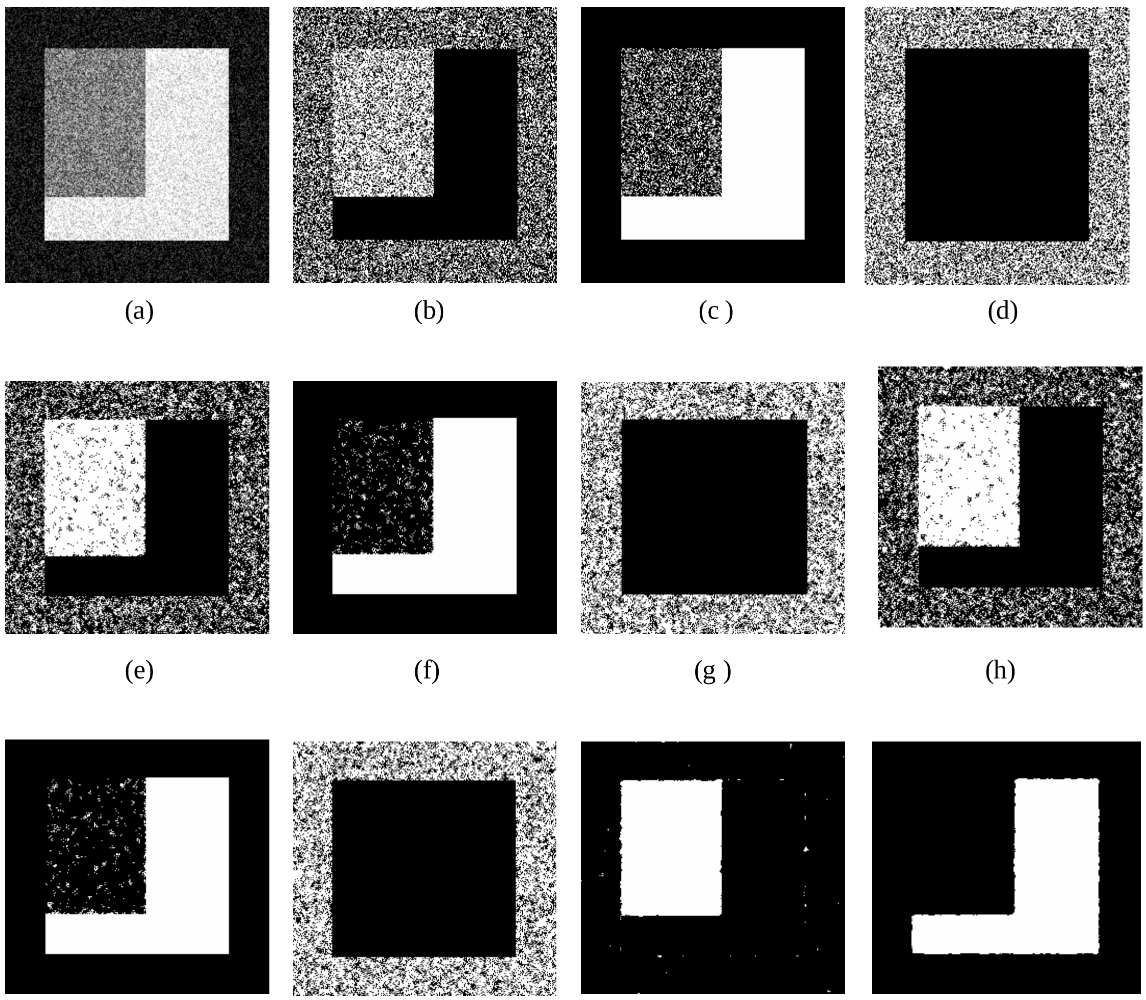
(a) (b) (c) (d)

(e) (f) (g) (h)

(i) (j) (k) (l)

(m)

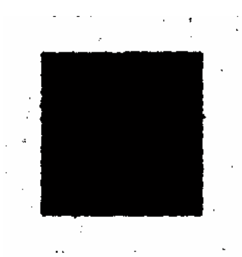
Figure 6.27 (a) Synthetic image with (-70,70) noise (b) – (d) First, second and third cluster using FCM (e) – (g) First, second and third cluster using sFCM (h) – (j) First, second and third cluster using MSFCM (k) – (m) First, second and third cluster using NAFCM



(a) (b) (c) (d)

(e) (f) (g) (h)

(i) (j) (k) (l)



(m)

Figure 6.28 (a) Synthetic image with (-90,90) noise (b) – (d) First, second and third cluster using FCM (e) – (g) First, second and third cluster using sFCM (h) – (j) First, second and third cluster using MSFCM (k) – (m) First, second and third cluster using NAFCM

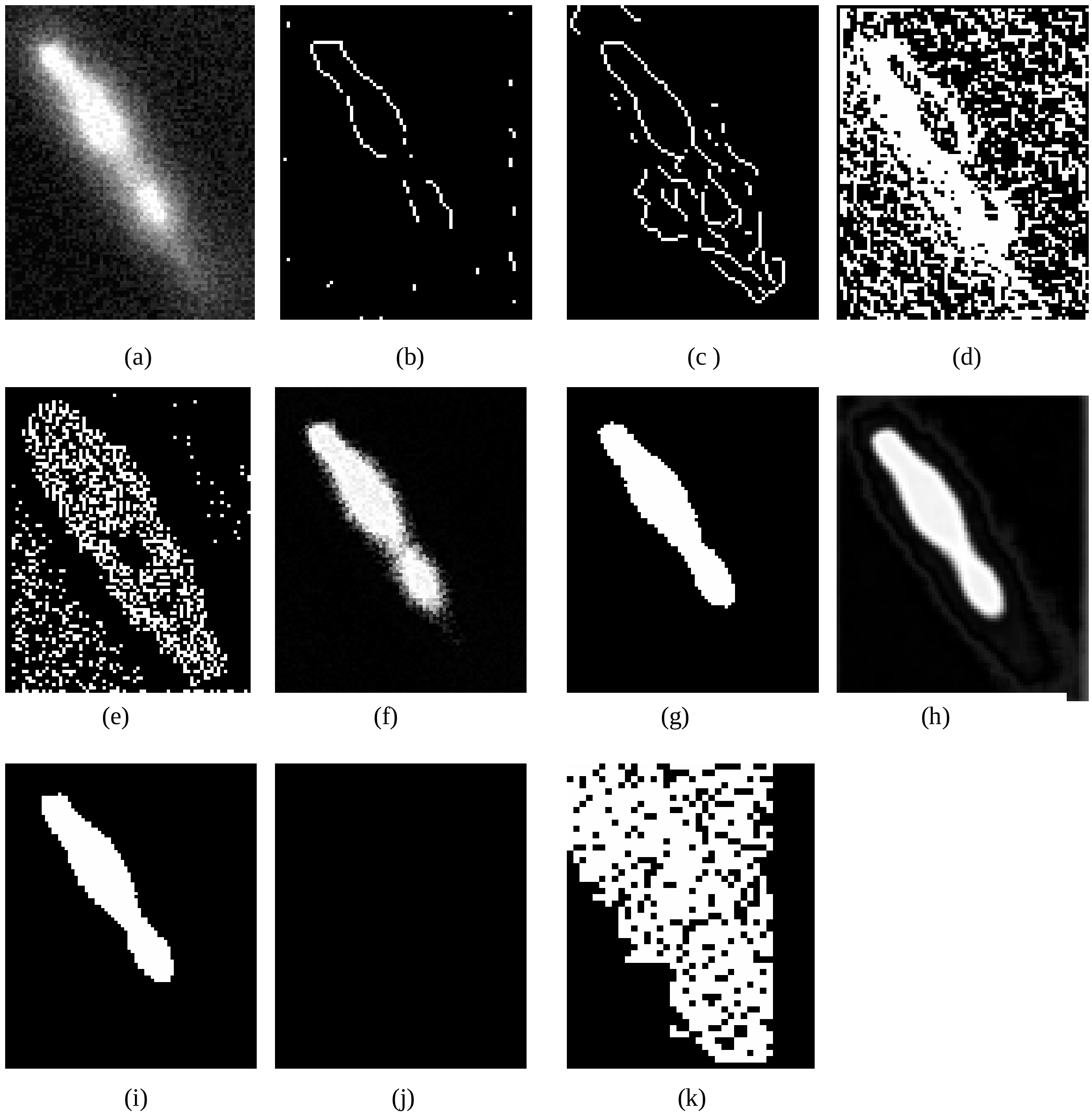
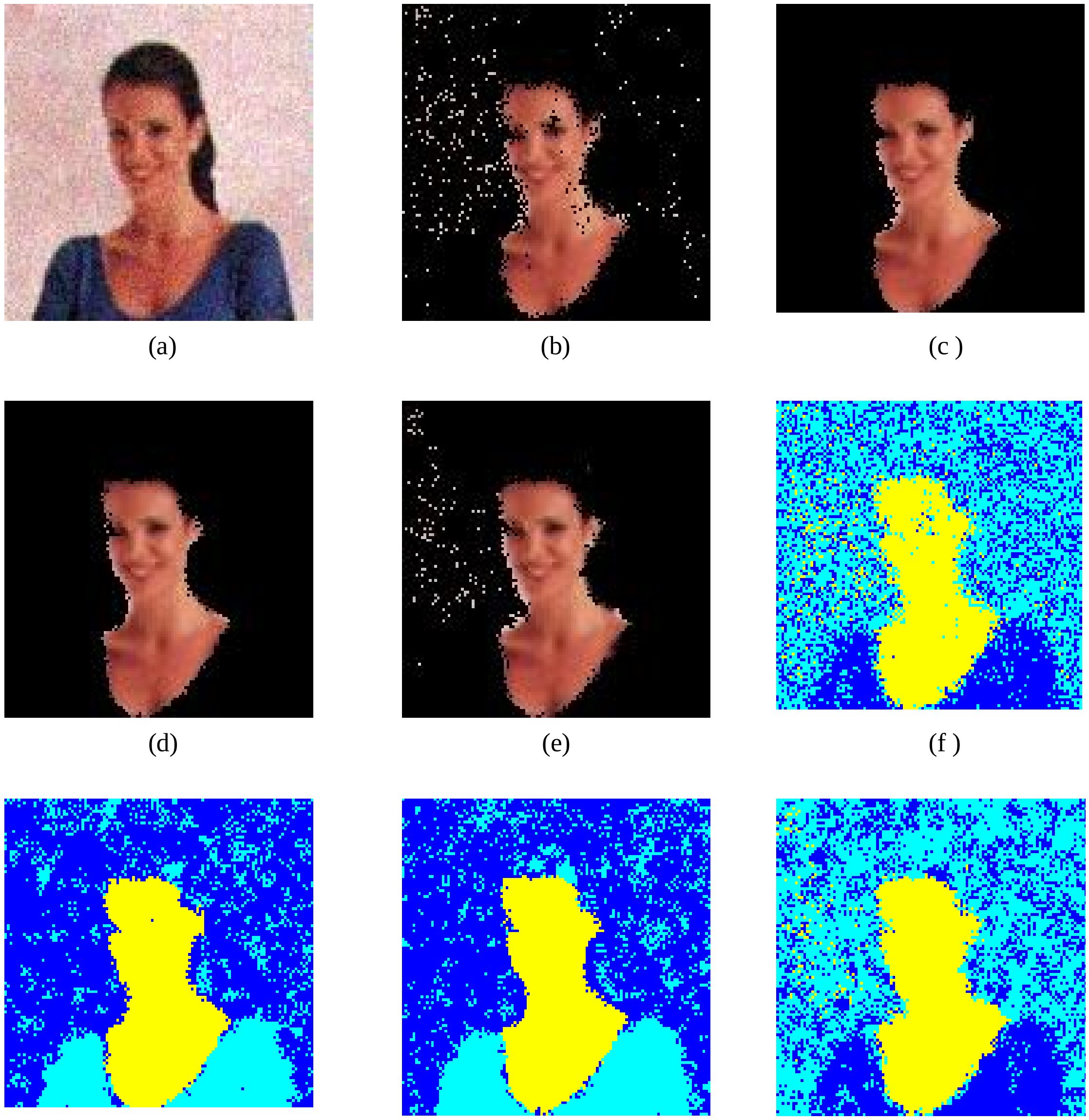


Figure 6.29 (a) Image of a weld crack with noise of (-35,35) (b) Result of LOG operator (c) Result of Canny edge detector (d) Result of FIS (e) Result of FMFED (f) Result of FCM (g) Result of sFCM (h) Result of MSFCM (i) Result of NAFCM (b) Result of region growing (c) Result of region split and merge

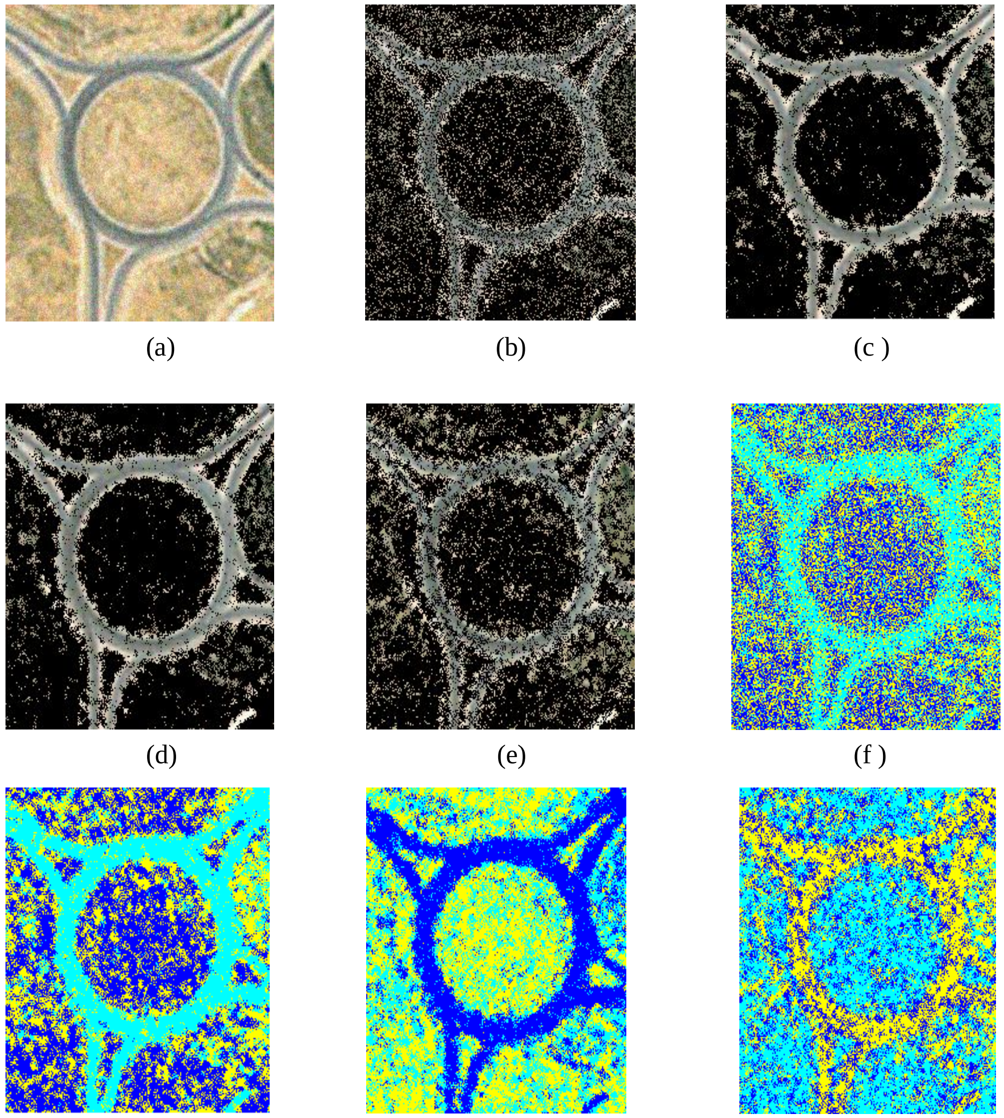


(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Fig. 6.30. Comparison of segmentation results on a human image corrupted with a noise varying between (-35,35). (a) Image with (-35,35) noise (b)-(d) Clustered image using FCM, sFCM and MSFCM resp. (e)-(g) Segmented image using FCM, sFCM and MSFCM respectively.

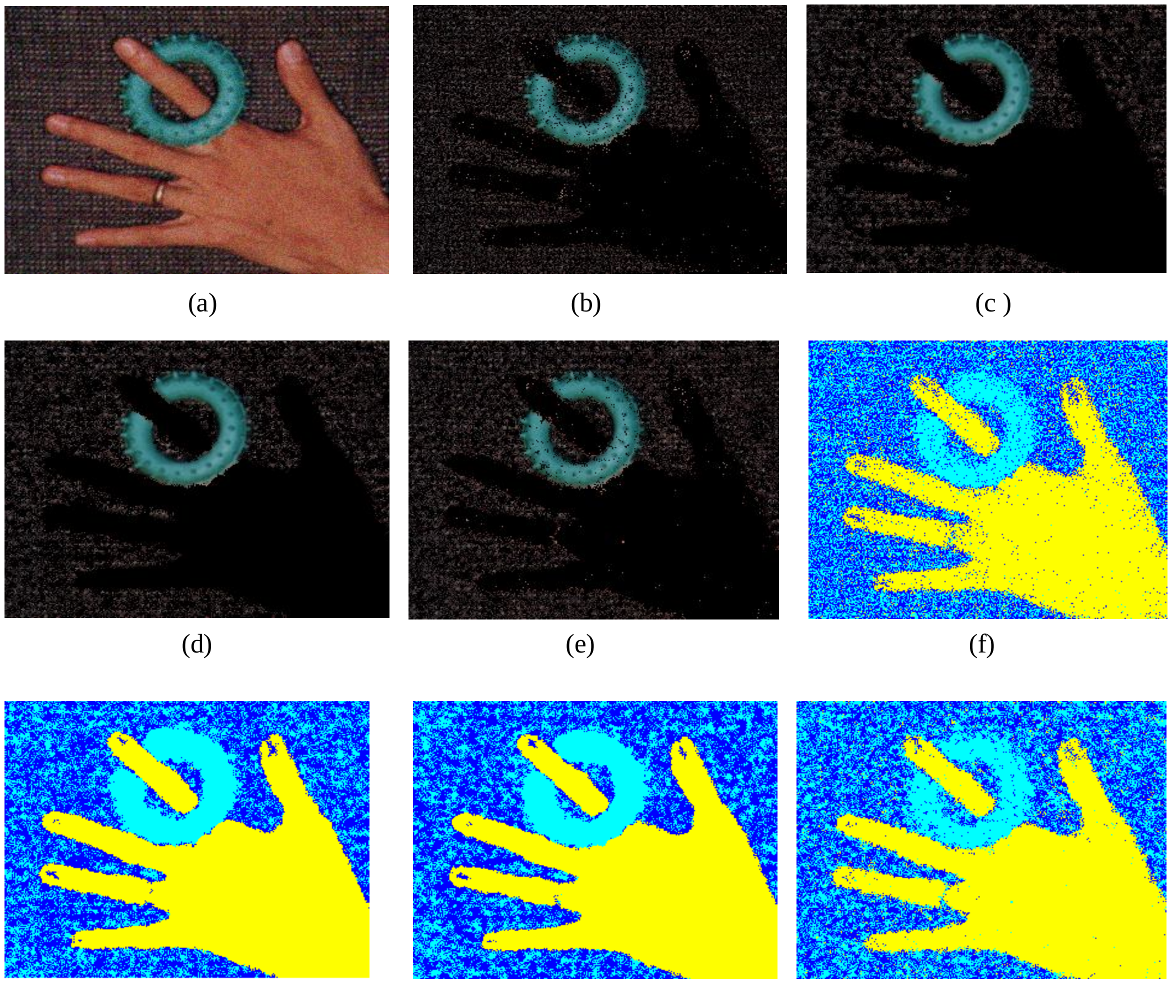


(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Fig.6.31. Comparison of segmentation results on a crossroad image corrupted with a noise varing between (-35,35). (a) Image with (-35,35) noise (b)-(e) Clustered image using FCM, sFCM , MSFCM and NAFCM resp. (f)-(i) Segmented image using FCM, sFCM, MSFCM and NAFCM respectively.

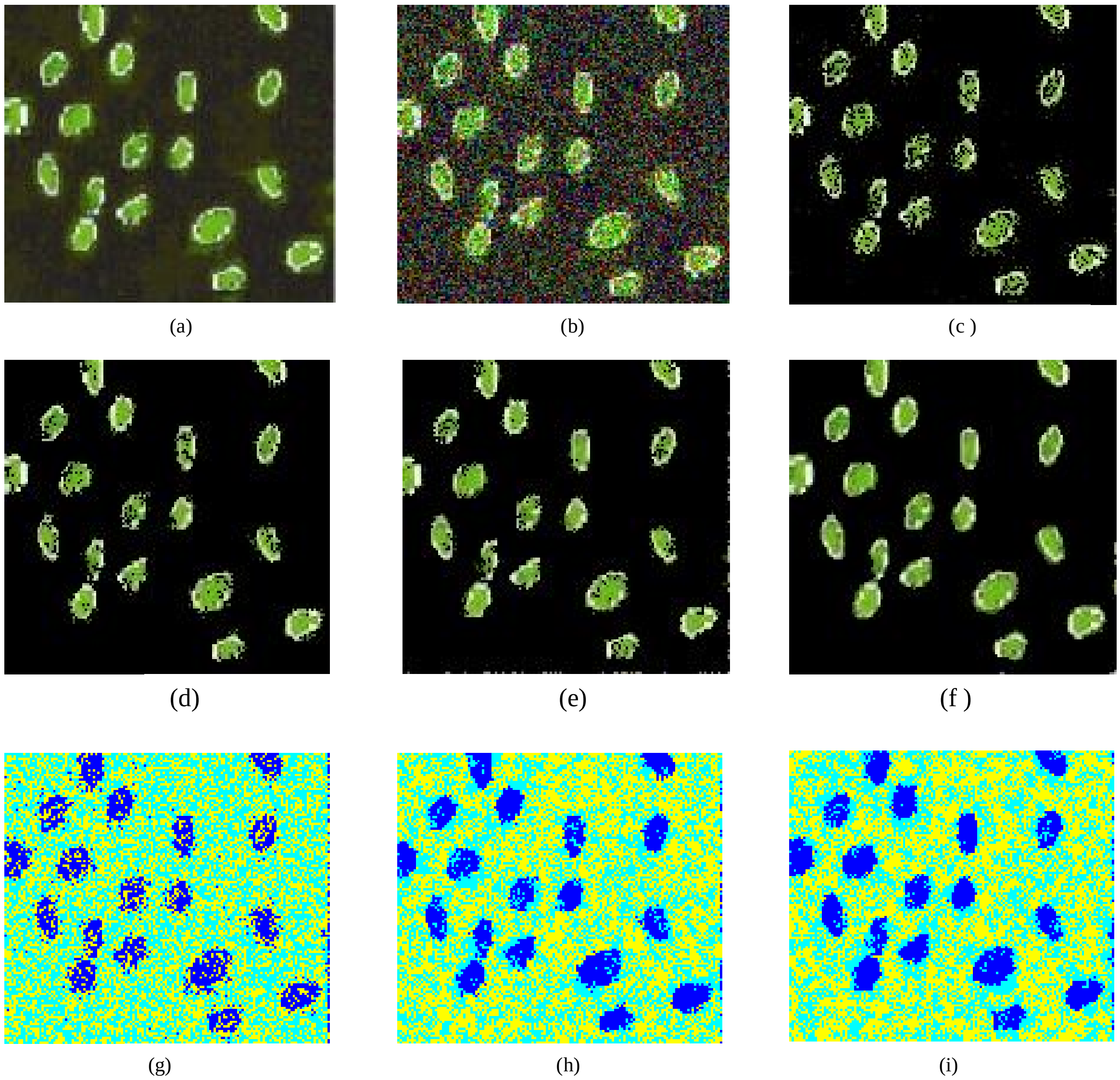


(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Fig.6.32. Comparison of segmentation results on a hand image corrupted with noise varying between (-45, 45). (a) Image with (-45,45) noise (b)-(e) Clustered image using FCM, sFCM, MSFCM and NAFCM resp. (f)-(i) Segmented image using FCM, sFCM, MSFCM and NAFCM respectively.



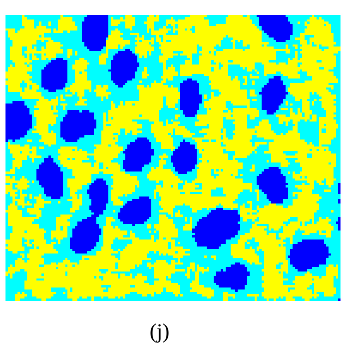
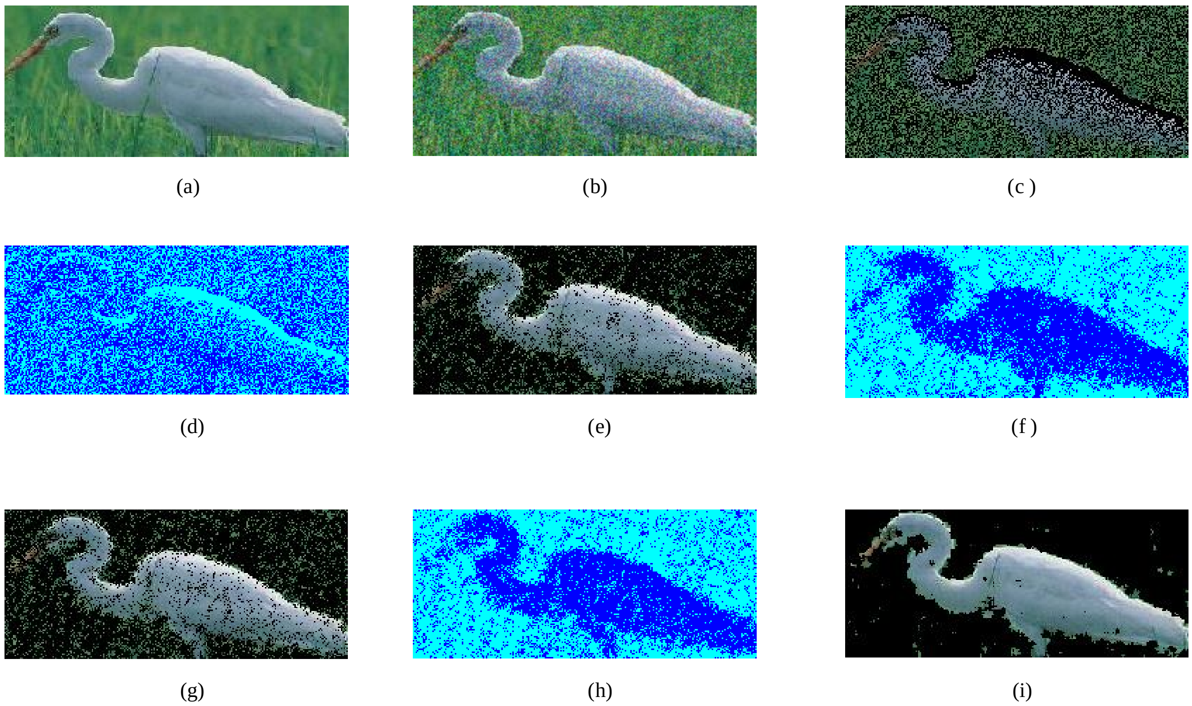


Fig.6.33. Comparison of segmentation results on a bacteria image corrupted with uniform random noise varying between (-90,90) . (a) Original bacteria image (b) Image with (-90,90) noise (c)-(f) Clustered image using FCM, sFCM, MSFCM and NAFCM resp. (g)-(i) Segmented image using FCM, sFCM, MSFCM and NAFCM respectively.



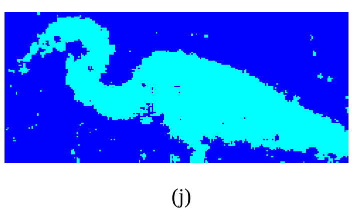
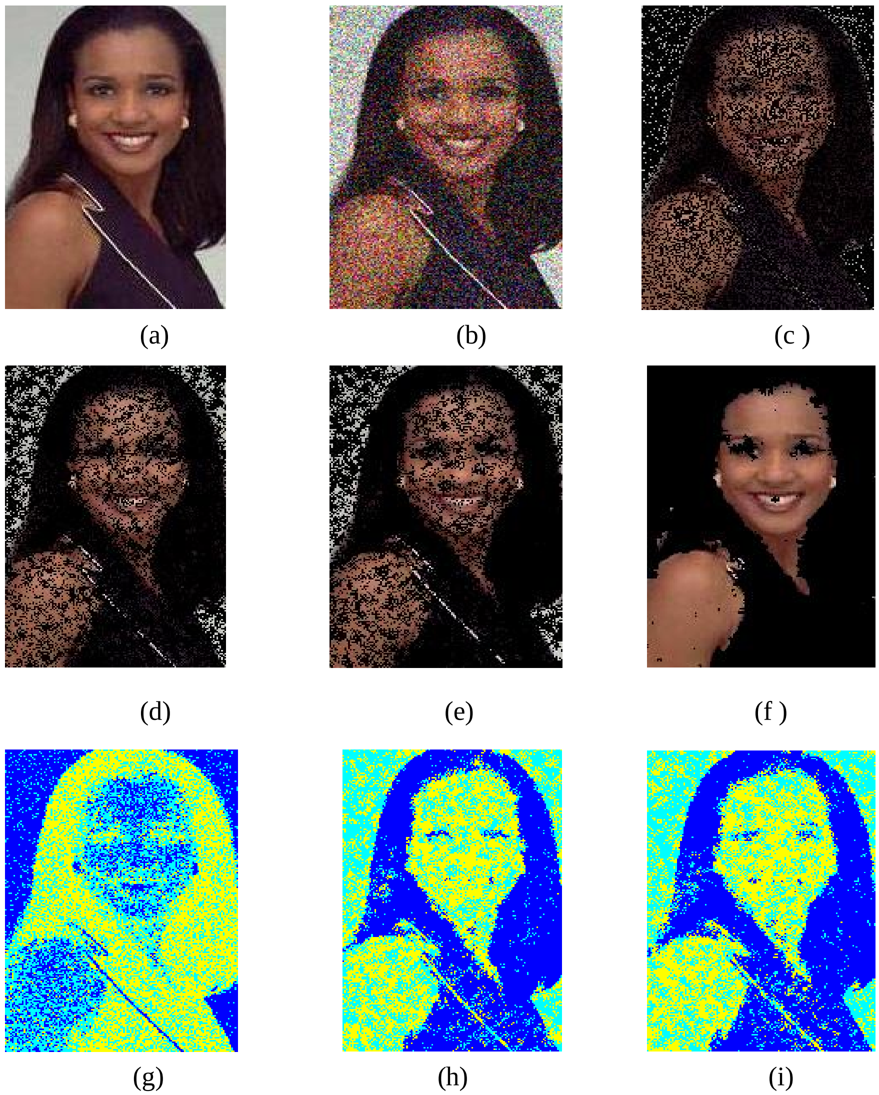


Fig.6.34. Comparison of segmentation results on a stork image corrupted with uniform random noise varying between (-60,60) . (a) Original bacteria image (b) Image with (-60,60) noise (c)-(f) Clustered image using FCM, sFCM, MSFCM and NAFCM resp. (g)-(i) Segmented image using FCM, sFCM, MSFCM and NAFCM respectively.



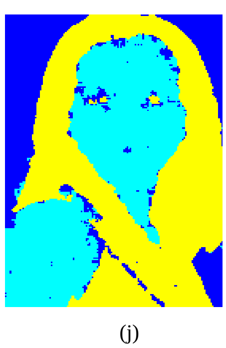


Fig.6.35. Comparison of segmentation results on a human image corrupted with uniform random noise varying between (-90,90) . (a) Original bacteria image (b) Image with (-90,90) noise (c)-(e) Clustered image using FCM,sFCM, MSFCM and NAFCM resp. (f)-(h) Segmented image using FCM, sFCM, MSFCM and NAFCM respectively.

Table 6.1: Comparision between various fcm based clustering methods with varying uniform random noise in RGB color space for hand image.

	Noise = (-35,35)		Noise= (-45,45)		Noise= (-60,60)		Noise = (-90,90)	
Clustering methods	V_{pc}	V_{pe}	V_{pc}	V_{pe}	V_{pc}	V_{pe}	V_{pc}	V_{pe}
FCM	0.6190	0.6684	0.5732	0.7349	0.5253	0.8090	0.4737	0.8898
sFCM	0.6203	0.6644	0.5632	0.7361	0.5246	0.8098	0.4761	0.8864
MSFCM	0.6206	0.6643	0.5644	0.7349	0.5266	0.8072	0.4768	0.8854
NAFCM	0.6294	0.6638	0.5790	0.7353	0.5249	0.8024	0.4783	0.8847

Table 6.2: Comparision between various fcm based segmentation techniques with varying uniform random noise in LAB color space for hand image

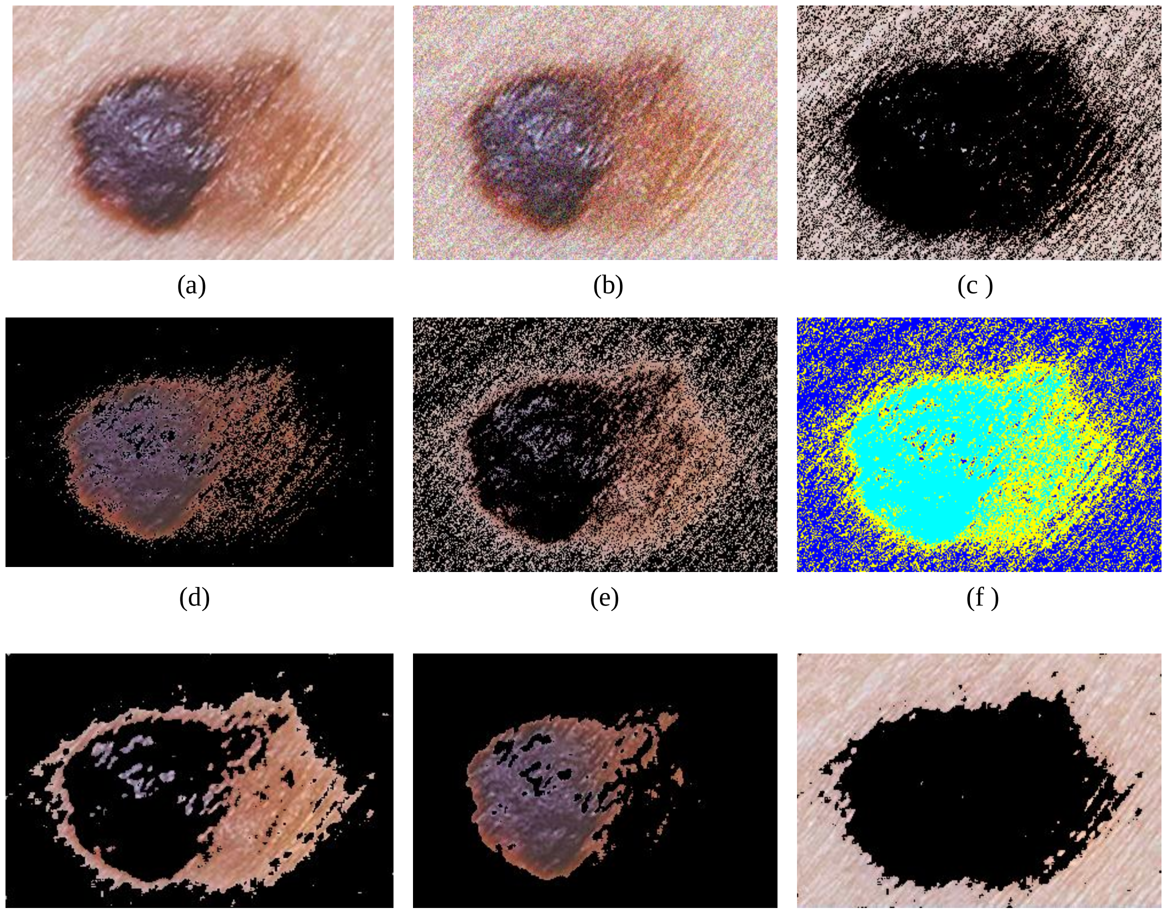
	Noise = (-35,35)		Noise = (-45,45)		Noise = (-60,60)		Noise = (-90,90)	
Clustering methods	V_{pc}	V_{pe}	V_{pc}	V_{pe}	V_{pc}	V_{pe}	V_{pc}	V_{pe}
FCM	0.6256	0.6582	0.5444	0.7796	0.5561	0.7652	0.5372	0.7958
sFCM	0.8213	0.2644	0.8697	0.2876	0.7505	0.3361	0.7142	0.4906
MSFCM	**0.8862**	**0.1682**	0.8722	0.2682	0.7662	0.3282	0.7263	0.4782
NAFCM	0.8682	0.2543	0.8602	0.2704	0.7837	0.3107	**0.7614**	**0.4122**

Table 6.3: Comparision between the fcm based segmentation techniques with varying uniform random noise in LAB color space for various test images.

		(35,-35)		(45,-45)		(60,-60)		(90,-90)	
	Techniques	V_{pc}	V_{pe}	V_{pc}	V_{pe}	V_{pc}	V_{pe}	V_{pc}	V_{pe}
Hand image in figure 6.31(a)	FCM	0.6256	0.6582	0.5914	0.7116	0.5561	0.7652	0.5372	0.7958
	sFCM	0.8213	0.2644	0.8697	0.2876	0.7505	0.3361	0.7142	0.4906
	MSFCM	**0.8862**	**0.1682**	0.8722	0.2682	0.7662	0.3282	0.7263	0.4782
	NAFCM	0.8682	0.2543	0.8602	0.2704	0.7837	0.3107	**0.7614**	**0.4122**
Cross-road image in figure 6.32(a)	FCM	0.5627	0.6508	0.5444	0.6796	0.5365	0.6922	0.5112	0.7836
	sFCM	0.7807	0.2268	0.7727	0.2298	0.6711	0.4889	0.5256	0.5213
	MSFCM	**0.7931**	**0.2129**	0.7784	0.2188	0.6720	0.4847	0.5388	0.5208
	NAFCM	0.7623	0.3315	0.7332	0.3781	0.6965	0.4406	**0.6918**	**0.4509**
Bacteria image in figure 6.33(a)	FCM	0.8772	0.2258	0.8316	0.2919	0.7500	0.4018	0.7468	0.4069
	sFCM	0.9870	0.0563	0.9754	0.1101	0.9012	0.1617	0.8016	0.3378
	MSFCM	**0.9896**	**0.0421**	0.9782	0.1121	0.9041	0.1576	0.8123	0.3268
	NAFCM	0.9621	0.1162	0.9299	0.1217	0.9139	0.3143	**0.8721**	**0.2193**
Bird image in figure 6.34(a)	FCM	0.9186	0.1104	0.9273	0.1483	0.8872	0.2122	0.7274	0.4317
	sFCM	0.9882	0.0129	0.9773	0.0364	0.9378	0.1542	0.8538	0.3413
	MSFCM	**0.9986**	**0.0131**	0.9784	0.0482	0.9390	0.1531	0.8674	0.3265
	NAFCM	0.9699	0.0528	0.9693	0.0538	0.9685	0.0856	**0.9364**	**0.1201**
Woman image in figure 6.35(a)	FCM	0.7766	0.4236	0.7158	0.5159	0.6455	0.6173	0.6220	0.6635
	sFCM	0.9846	0.0573	0.9011	0.1891	0.8618	0.2261	0.7981	0.4509
	MSFCM	**0.9924**	**0.0121**	0.9268	0.1383	0.8735	0.2213	0.8184	0.3870
	NAFCM	0.9778	0.0892	0.9149	0.1191	0.8868	0.1208	**0.8691**	**0.2498**

Table : 6.4 : Number of misclassified pixels with FCM,sFCM,MSFCM and NAFCM for synthetic image shown in figure 6.2(a) with different noise values

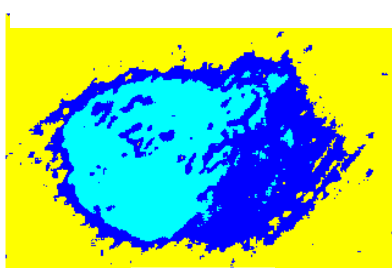
Noise	CLASS	Clustering methods			
		FCM	sFCM	MSFCM	NAFCM
		Percentage of misclassified pixels in cluster 1, 2 and 3			
(-40,40)	1	0.1058	0	**0**	0.0677
	2	0.0088	0	**0**	0.0147
	3	0.0665	0.0055	**0**	0.0665
(-60,60)	1	16.9915	0.0055	0.0021	0
	2	3.7992	0.0021	0	0
	3	0.7594	0.0055	0.0055	0.0055
(-70,70)	1	58.8833	32.5532	29.769	0.0973
	2	30.8202	13.0897	11.9596	0.2619
	3	17.8271	0.6763	0.5876	1.2417
(-80,80)	1	67.339	65.7837	63.2854	2.7843
	2	37.2591	27.0710	25.8174	1.2007
	3	21.4911	2.4224	2.0067	1.5854
(-90,90)	1	712.7775	76.7477	77.1239	**5.0418**
	2	99.9117	33.4922	32.0679	**1.2919**
	3	23.2761	6.5466	3.5477	**1.5355**



(a) (b) (c)

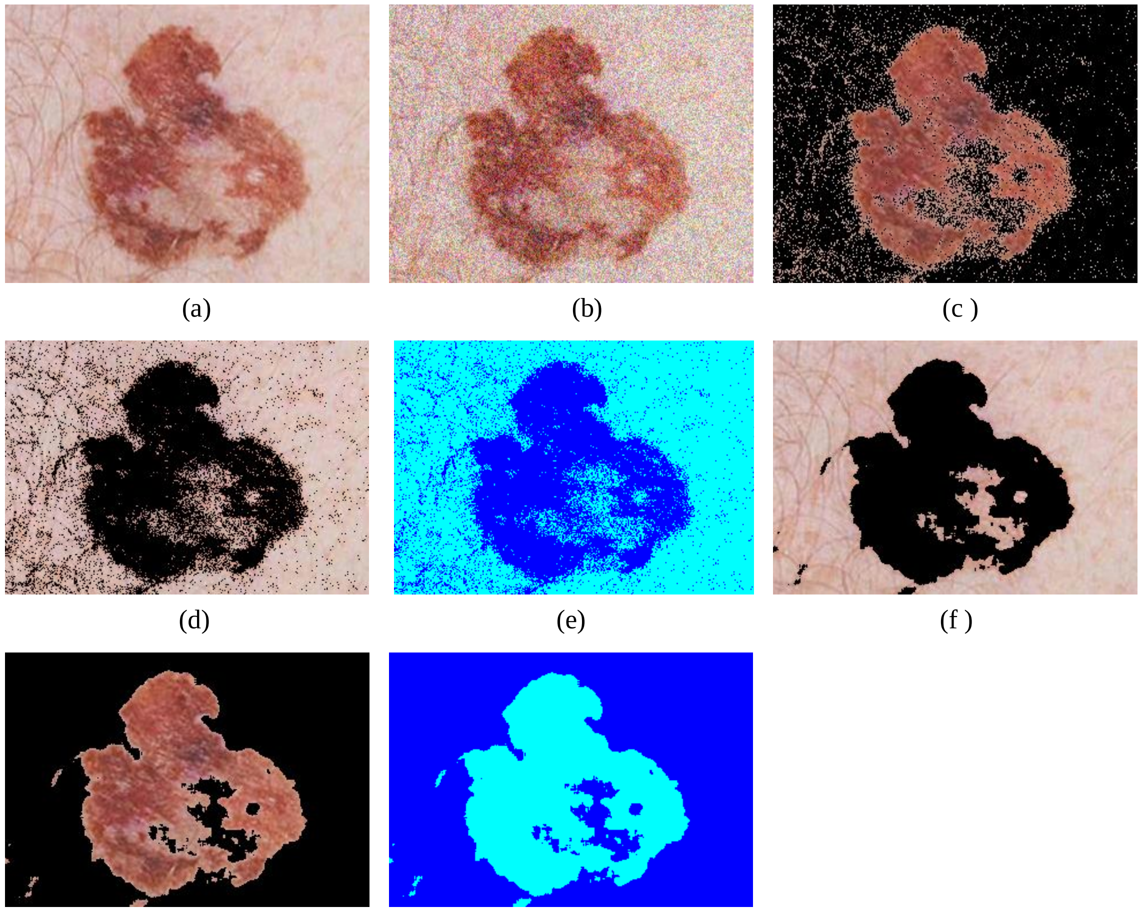
(d) (e) (f)

(g) (h) (i)



(j)

Figure 6.36. (a) Original image (b) Image with (-60,60) noise (c)-(e) Object in 1st , 2nd & 3rd cluster respectively using FCM (f) Segmented image using FCM (g)-(i) Object in 1st ,2nd & 3rd cluster using proposed NAFCM (j) Segmented image using NAFCM.



(a) (b) (c)

(d) (e) (f)

(g) (h)

Figure 6.37. (a) Original image (b) Image with (-60,60) noise (b)-(c) Object in 1st , 2nd & 3rd cluster respectively using FCM (d) Segmented image using FCM (e)-(f) Object in 1st & 3rd cluster using NAFCM (f) Segmented image using NAFCM.

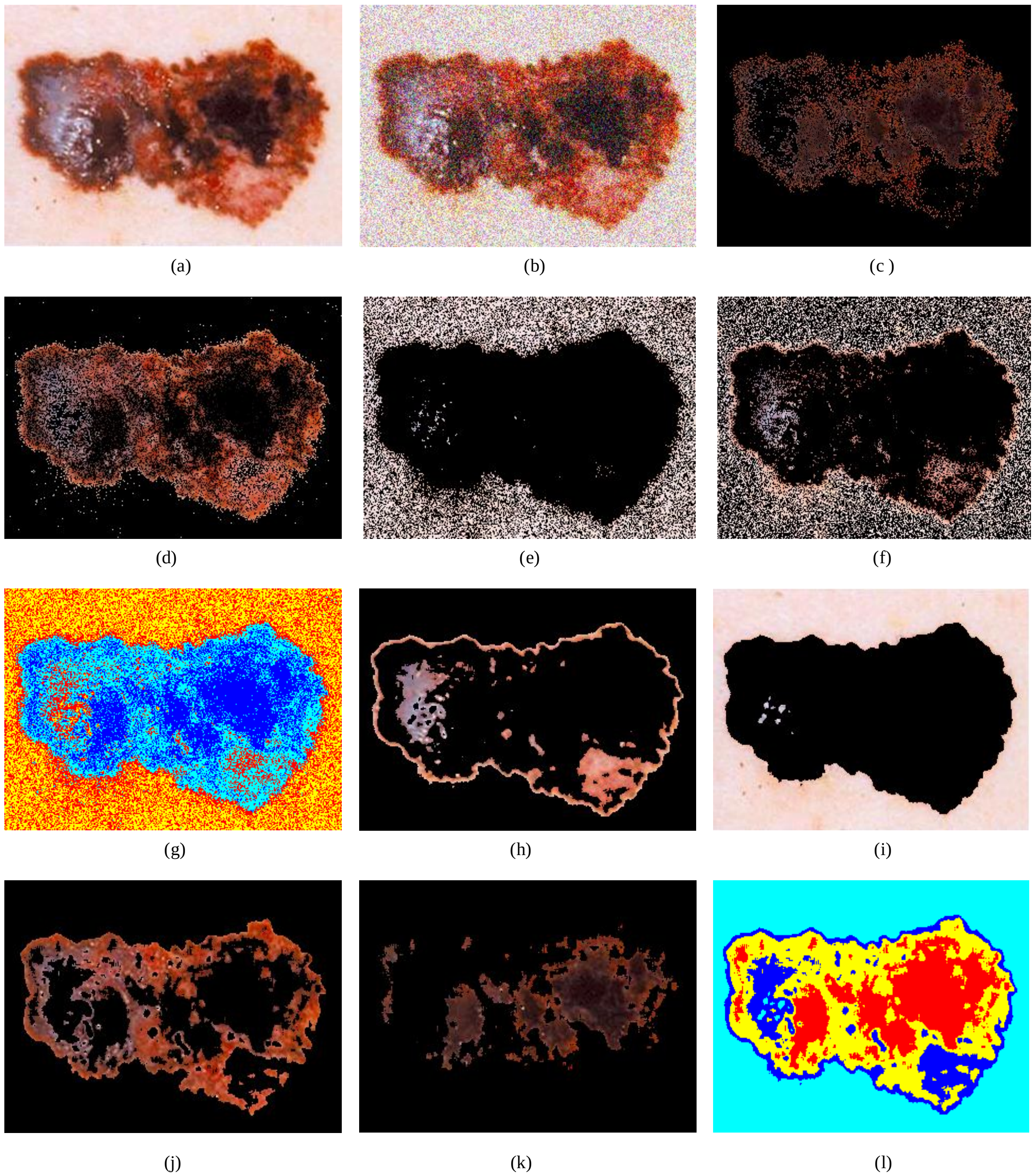


Figure 6.38. (a)Original image (b) Image with (-90,90) noise (c)-(f) Object in 1^{st} , 2^{nd} , 3^{rd} & 4^{th} cluster respectively using FCM (g) Segmented image using FCM (h)-(k) Object in 1^{st} ,2^{nd} , 3^{rd} & 4^{th} cluster using NAFCM (l) Segmented image using NAFCM.

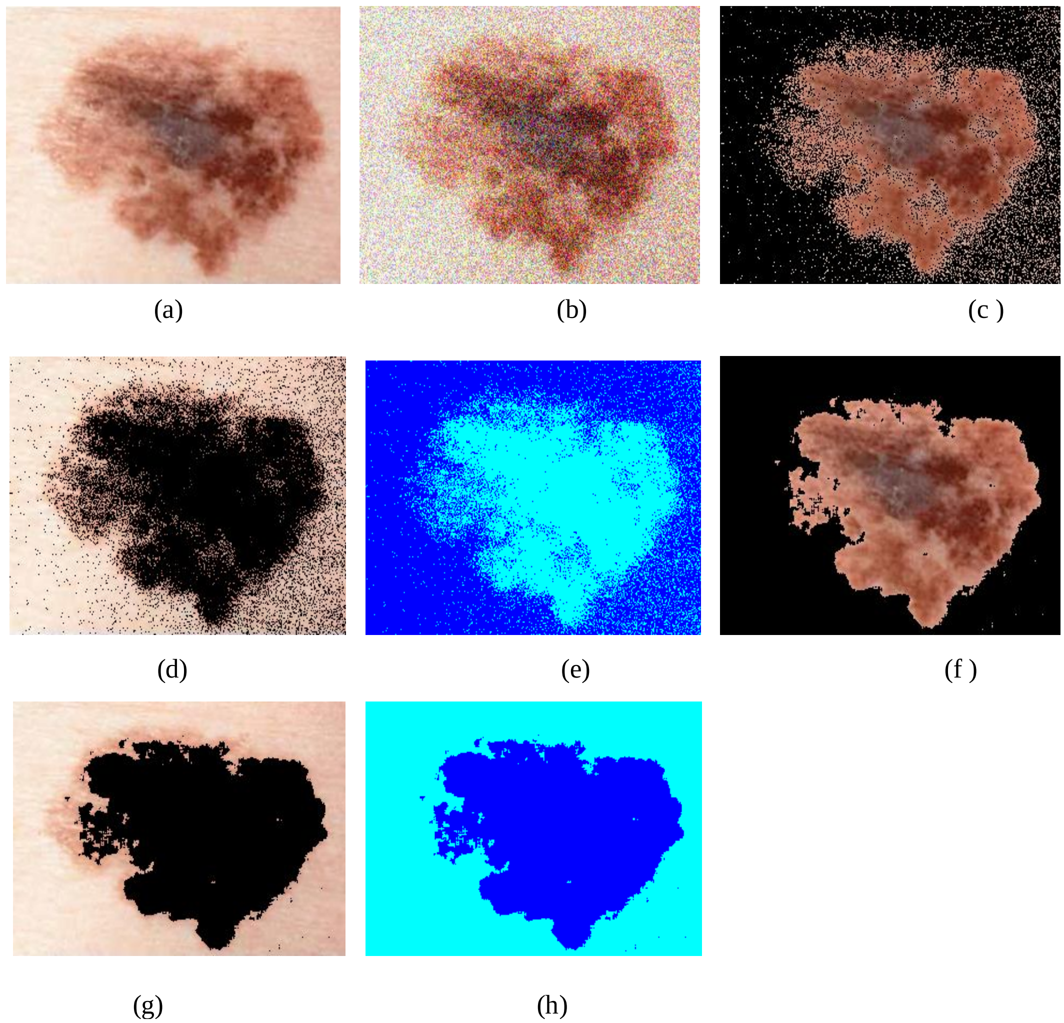
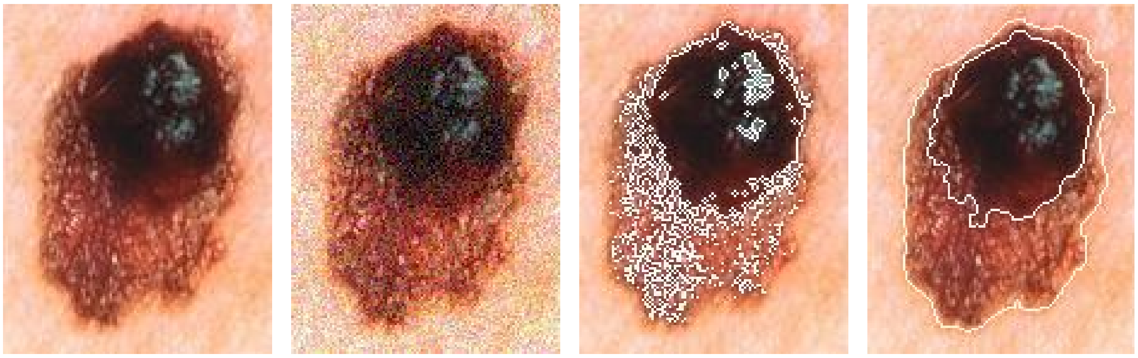
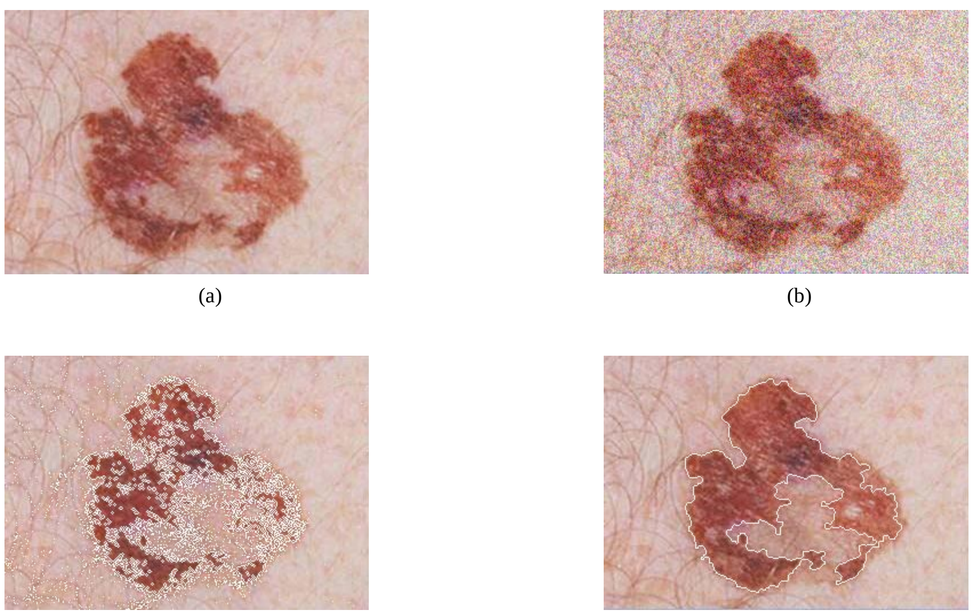


Figure 6.39. (a)Original image (b) Image with (-90,90) noise (c)-(d) Object in 1^{st} & 2^{nd} cluster respectively using FCM (e) Segmented image using FCM (f)-(g) Object in 1^{st} & 2^{nd} cluster using NAFCM (h) Segmented image using NAFCM.



(a) (b) (c) (d)

Figure 6.40. (a) Original image (b) Image with (-60,60) noise (c) Skin lesion segmented with FCM technique after addition of noise (d) Skin lesion segmented using NAFCM after addition of noise.



(a) (b) (c) (d)

Figure 6.41. (a) Original image (b) Image with (-60,60) noise (c) Skin lesion segmented with FCM technique after addition of noise (d) Skin lesion segmented using NAFCM after addition of noise.

Table 6.5 Effect of increasing noise on the various tumor test images

		(35,-35)		(45,-45)		(60,-60)		(90,-90)	
Tumor image in	Techniques	V_{pc}	V_{pe}	V_{pc}	V_{pe}	V_{pc}	V_{pe}	V_{pc}	V_{pe}
figure 6.36(a)	FCM	0.6520	0.6089	0.6056	0.6783	0.5599	0.7487	0.4963	0.8515
	NAFCM	0.8424	0.2067	0.8206	0.2403	0.8162	0.2448	0.8081	0.3204
figure 6.37(a)	FCM	0.6237	0.6450	0.5847	0.7073	0.5384	0.7818	0.4816	0.8752
	NAFCM	0.8171	0.3407	0.8028	0.3605	0.7932	0.3725	0.7623	0.4255
figure 6.38(a)	FCM	0.7729	0.4203	0.7397	0.4779	0.6838	0.5682	0.5844	0.7159
	NAFCM	0.8907	0.2126	0.8858	02216	0.8745	0.2329	0.8568	0 .2399
figure 6.39(a)	FCM	0.6880	0.5557	0.6496	0.6155	0.5992	0.6927	0.5296	0.7998
	NAFCM	0.8541	0.2858	0.8389	0.3109	0.8202	0.3416	0.7831	0.3967

6.3 Conclusion

In **chapter 2,** various basic methods of image segmentation have been studied. It is observed from Fig. 6.1- Fig 6.6 that edge detectors are quite simple to execute and they are able to find the edges of objects present in all kind of images. Whereas, the region based methods are application dependent and the condition for region growing or region splitting and merging may change from one image to another. Hence the fuzzy edge detection based segmentation algorithms are explored in the next chapter.

In **chapter 3** two fuzzy methods for edge detection based segmentation are studied, the conclusion regarding the two methods is as follows:

FIS for edge detection based segmentation:

From the simulation results in Fig. 6.7-Fig 6.9 it can be very easily concluded that the FIS system developed better than the popular Sobel but its results are not as good as Canny operator. But one of the main problems with implementing such a FIS system is the amount of time required during processing. One of the main problems in implementing a FIS system is the amount of time required during processing. In addition to that, despite being used in a wide range of applications, both the structure of membership functions and derivation of their relevant parameters were still very much application domain and image dependent.

Multilevel fuzzy edge detector for segmentation:

This method has clear advantage over the rule based method as it does not involve changing the structure of membership function according to a particular application. This method gives better edges as compared to Sobel and Canny edge detector as seen in Fig.6.10- Fig.6.14 and it is also much faster as compared to the FIS algorithm for edge detection.

The edge detection algorithms are normally followed by edge linking procedures to bridge gaps in region boundary. We apply simple morphological tools for the edge linking problem. The results of applying edge linking by morphological operators on the edge detected image, is shown in Fig 6.15. The edge detection method considered for all these images is FMFED algorithm. The reason for not applying FIS based edge detector is its poor quality of edge detection compared to some older techniques like canny edge detector. The algorithm is also much faster as compared to the FIS algorithm for edge detection

In **chapter 4**, algorithms for segmentation of color images using fuzzy clustering have been developed

The segmentation method described in section 4.4 uses Fuzzy c-means as a tool for segmentation. The simulation results in Fig. 6.16-Fig. 6.21, show how the FCM is applied for segmentation. By observing the results it can be said that FCM can be successfully applied for clustering based segmentation of different types of images.

The algorithm described in section 4.5 applies, enhancement algorithm (CLAHE) to FCM. The enhancement is applied only to the Luminosity layer (L) as it is the layer containing information about brightness of the image. By first enhancing the image and then performing clustering we are able to extract quite good segmentation results. The CLAHE algorithm spreads the brightness uniformly among all the pixels hence too bright pixels does not remain too bright and pixels having low brightness value are made to have more high brightness information.

From the simulation results in chapter 6 (Fig. 6.22-Fig 6.25) it can be said that: CLAHEFCM improves the clustering and hence segmentation results of images which are not evenly illuminated.

In **chapter 5**, algorithms for segmentation by incorporating spatial property of pixels in fuzzy clustering have been developed. The algorithm described in section 5.2 presents a modified spatial FCM algorithm (MSFCM) and observes its effect on color images degraded by random noise. The algorithm was realized by modifying the spatial function as described above. Qualitative (Fig.6.26-Fig. 6.35) and quantitative experimental results (Table 6.3) show that the proposed MSFCM (highest V_{pc} and lowest V_{pe}) algorithm is superior to standard FCM, sFCM and NAFCM when the clustering is done under low noise condition.

The proposed method (NAFCM) is an extension of FCM algorithm which takes into account the neighbourhood attraction of the pixels and observes its effect on segmentation of color images degraded by random noise. The algorithm is tested on a synthetic image (Fig. 6.26- Fig. 6.28), greyscale image (Fig. 6.29) and various other images (Fig. 6.30-Fig 6.35), having a noise of (-60,60) and higher . The results obtained by using proposed method have been compared with the results of other FCM based segmentation techniques. By observing the results (Fig. 2.26-Fig.6.35 and Table 6.3) it can be said that NAFCM (highest V_{pc} and lowest V_{pe}) gives the best clustering and hence segmentation result under high noise condition.

Since the objective function of standard FCM was not modified in both the proposed methods, as in case of most techniques applying FCM for segmentation, the inbuilt FCM function present in recent MATLAB versions can be very easily applied for problem related to clustering and segmentation while applying these two algorithms.

By observing Table 6.1 and Table 6.2 it can be said that clustering based segmentation performed in LAB color space gives higher values of V_{pc} and lower values for V_{pe} (condition for good clustering and segmentation) under increasing noise increasing from (-35,35) to (-90,90), as compared to clustering based segmentation in RGB color space.

From Table 6.4 it is observed that as noise increases the percentage of misclassified pixels for every class also increases. MSFCM based segmentation gives the least percentage of misclassified pixels under low noise condition. NAFCM based segmentation gives the least percentage of misclassified pixels for every class under high noise condition.

From Table 6.5 it can be concluded that NAFCM can be used for segmentation of melanoma images.

CHAPTER 7

Conclusion

7

Preview

In this research work various popular fuzzy techniques used for image segmentation available in the literature are studied. These fuzzy techniques can be combined with any other method to enhance the ability of the algorithm in good segmentation. However, due to the limitation of other fuzzy techniques, fuzzy clustering based segmentation has been considered in this thesis. One major limitation with FCM based segmentation is that it does not take into consideration the spatial context of the image pixels, due to this FCM clustering based segmentation is sensitive to noise and imaging artefacts.

Hence to compensate for this drawback of FCM clustering based segmentation, efforts have been made to develop algorithms, which are an extension to the standard FCM and take into account the spatial context of pixels. These algorithms are observed to perform well on noisy images. An FCM clustering algorithm for segmentation of images under uneven illumination has also been developed.

The performance of the proposed algorithms for segmentation has been compared with existing algorithms. The objective evaluation metric used for clustering based segmentation are partition coefficient and partition entropy. All algorithms have been compared with respect to their execution time.

7.1 Comparative Analysis

The execution time of various segmentation methods such as region based segmentation method, edge detection based segmentation method and clustering based segmentation method is compared in Table 7.1. The hardware platform used is a Pentium IV core 2 duo processor, 2.4GHz (clock), 2GB (RAM) with windows vista 64 bit operating system.

A qualitative comparison between various existing algorithms and FCM is done on a weld crack image. The results are shown in Fig. 7.1.
The existing and the proposed segmentation algorithms are simulated on a different color test image. The test image is corrupted with a noise varying between (-35, 35), (-45, 45), (-60, 60) and (-90, 90). The performance of various clustering methods is compared in terms of V_{pc} and V_{pe} in *LAB* color space (used in our algorithm) is shown in Table 7.2.

Table 7.1: Segmentation performance of various segmentation methods in terms of Execution time

	Segmentation methods	Execution time (sec)
Region based segmentation method	Region growing Region split and merge	5.78 5.906
Edge detection based segmentation method	LOG Canny FIS Multi-level fuzzy edge detector	0.985 1.781 13.328 1.593
Clustering based segmentation method	FCM CLAHEFCM sFCM MSFCM NAFCM	1.469 1.625 2.141 2.266 2.922

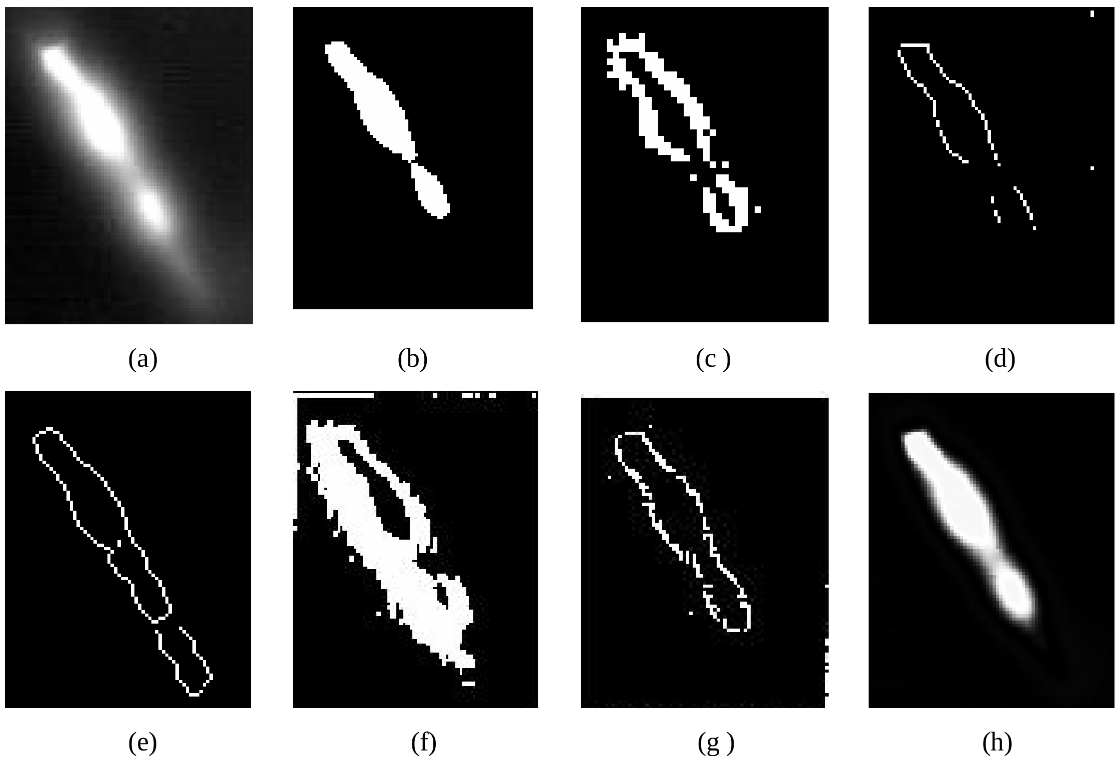


Figure. 7.1 (a) Image of a weld crack (b) result of region growing (c) Result of region split and merge (d) Result of LOG operator (e) Result of Canny edge detector (f) Result of FIS (g) Result of FMFED (h) result of FCM

Table 7.2: Comparision between the fcm based segmentation techniques with varying uniform random noise in LAB color space for various test images.

		(35,-35)		(45,-45)		(60,-60)		(90,-90)	
	Techniques	V_{pc}	V_{pe}	V_{pc}	V_{pe}	V_{pc}	V_{pe}	V_{pc}	V_{pe}
Hand image in figure 6.31(a)	FCM	0.6256	0.6582	0.5914	0.7116	0.5561	0.7652	0.5372	0.7958
	sFCM	0.8213	0.2644	0.8697	0.2876	0.7505	0.3361	0.7142	0.4906
	MSFCM	**0.8862**	**0.1682**	0.8722	0.2682	0.7662	0.3282	0.7263	0.4782
	NAFCM	0.8682	0.2543	0.8602	0.2704	0.7837	0.3107	**0.7614**	**0.4122**
Cross-road image in figure 6.32(a)	FCM	0.5627	0.6508	0.5444	0.6796	0.5365	0.6922	0.5112	0.7836
	sFCM	0.7807	0.2268	0.7727	0.2298	0.6711	0.4889	0.5256	0.5213
	MSFCM	**0.7931**	**0.2129**	0.7784	0.2188	0.6720	0.4847	0.5388	0.5208
	NAFCM	0.7623	0.3315	0.7332	0.3781	0.6965	0.4406	**0.6919**	**0.4509**
Bacteria image in figure 6.33(a)	FCM	0.8772	0.2258	0.8316	0.2919	0.7500	0.4018	0.7468	0.4069
	sFCM	0.9870	0.0563	0.9754	0.1101	0.9012	0.1617	0.8016	0.3378
	MSFCM	**0.9896**	**0.0421**	0.9782	0.1121	0.9041	0.1576	0.8123	0.3268
	NAFCM	0.9621	0.1162	0.9299	0.1217	0.9139	0.3143	**0.8722**	**0.2193**
Bird image in figure 6.34(a)	FCM	0.9186	0.1104	0.9273	0.1483	0.8872	0.2122	0.7274	0.4317
	sFCM	0.9882	0.0129	0.9773	0.0364	0.9378	0.1542	0.8538	0.3413
	MSFCM	**0.9986**	**0.0131**	0.9784	0.0482	0.9390	0.1531	0.8674	0.3265
	NAFCM	0.9699	0.0528	0.9693	0.0538	0.9685	0.0856	**0.9364**	**0.1201**
Woman image in figure 6.35(a)	FCM	0.7766	0.4236	0.7158	0.5159	0.6455	0.6173	0.6220	0.6635
	sFCM	0.9846	0.0573	0.9011	0.1891	0.8618	0.2261	0.7981	0.4509
	MSFCM	**0.9924**	**0.0121**	0.9268	0.1383	0.8735	0.2213	0.8184	0.3870
	NAFCM	0.9778	0.0892	0.9149	0.1191	0.8868	0.1208	**0.8692**	**0.2498**

7.2 Conclusion

In Table 7.1 it is observed that LoG has the least execution time among all segmentation methods considered as it is the simplest method. Among the clustering based segmentation method, the standard FCM based segmentation takes the least execution time.

It is observed from Fig. 7.1 that FCM based segmentation gives the best segmentation result without noise. The segmentation by FCM clustering gives a smooth contour of the weld crack as compared to other methods.

By observing Table 6.1 and Table 6.2 it can be said that clustering based segmentation performed in LAB color space gives higher values of V_{pc} and lower values for V_{pe} (condition for good clustering and segmentation) under increasing noise increasing from (-35,35) to (-90,90), as compared to clustering based segmentation in RGB color space.

Hence it can be concluded that for real life images, FCM based segmentation in LAB color space gives better results as compared to RGB color space.

By observing Fig. 6.26 - Fig 6.35 and Table 7.2, it can be said that the proposed MSFCM gives the best clustering(highest V_{pc} and lowest V_{pe}) result under low noise condition as compared to FCM, sFCM or NAFCM. It is also observed that NAFCM gives the best clustering (highest V_{pc} and lowest V_{pe}) and hence segmentation result under high noise condition.

Hence it is concluded that MSFCM gives the best clustering and hence segmentation result under low noise condition and NAFCM gives the best clustering and segmentation result under high noise condition.

From Table 6.4 it is observed that as noise increases the percentage of misclassified pixels for every class also increases. **MSFCM based segmentation gives the least percentage of misclassified pixels under low noise condition. NAFCM based segmentation gives the least percentage of misclassified pixels for every class under high noise condition.**

7.3 Scope of Future work

1. The execution time of the proposed method is an area of concern. Hence clustering methods which are less time consuming can be developed for segmentation.

2. The fuzzy clustering based method can be combined with other methods like Genetic algorithm and Level set methods to give better segmentation results.

3. The number of cluster has to be fixed initially in FCM based segmentation methods. Some method which doesn’t require fixing of number of clusters before clustering can also be used for segmentation.

REFERENCES

[1] Law Todd, Itoh Hidenori, et al. Image filtering, edge detection, and edge tracing using fuzzy reasoning, *IEEE transactions on pattern analysis and machine intelligence*, vol. 18, no. 5, May(1996), pp. 481-491.

[2] Russo F. Edge detection in noisy images using fuzzy reasoning, *IEEE transactions on instrumentation and measurement,* vol.47, no.5 1998, pp. 1102-1105.

[3] Akbari Sheikh A. and Soraghan J.J. Multiscale fuzzy reasoning for automatic object extraction, *Pattern recognition letters*, vol. 26, no. 1, January(2005), pp. 77-81.

[4] Vlachos, I.K. and Sergiadis, G.D. Fuzzy reasoning scheme for edge detection using local edge information based on Renyi's entropy, *Proceedings of seventh international symposium on signal processing and its applications*. Vol.1, July (2003), pp. 549-552.

[5] Bellon Olga Regina Pereira, Dhirene Alexandre Ibrahim et al. Edge detection to guide image segmentation by clustering techniques, *International conference on image processing,* Vol. 2(1999), pp. 725-729.

[6] Xiaohan Yu, J. Yla-Jaaski et al. A new algorithm for texture segmentation based on edge detection, *Pattern recognition* , vol. 24, no. 11, 1991,pp. 1105-1112.

[7] Yi Liu, Xue-quan Chen. An edge detection algorithm of remote sensing images based on fuzzy sets, *International conference on communications, circuits and system*, vol. 2, June 2004, pp. 984-998.

[8] Wu Jinbo, Yin Zhouping, and Xiong Youlun. The fast multilevel fuzzy edge detection of blurry images, *IEEE signal processing letters,* vol. 14, no. 5, May (2007),pp. 344-347.

[9] Miosso Cristiano Jacques and Adolfo Bauchspiess. Fuzzy inference system applied to edge detection in digital images, *Proceedings of the V Brazilian conference on neural networks, April(2001), pp. 481-486.*

[10] HT Farrah Wong, Nagarajan Ramachandran et al. An image segmentation method using fuzzy-based threshold, *International symposium on signal processing and its applications (ISSPA)*, August (2001), pp. 144-147.

[11] Tao Wen-Bing, Tin Jin-Win et al. Image segmentation by three-level thresholding based on maximum fuzzy entropy and genetic algorithm, *pattern recognition letters*, vol. 24, June(2003),pp.3069-3078.

[12] Cheriet M., Said J.N. and Suen C.Y. A recursive thresholding technique for image segmentation, *IEEE transactions on image processing*, vol. 7, no.6, June 1998, pp. 918-921.

[13] Sahoo P. K., Soltani S., and Wong A. K. C. A survey of thresholding techniques, *Comput. Vision Graph. Image Processing*, vol. 41, pp. 233-260, 1988.

[14] Jain A. K., Fundamentals of digital image processing, *Prentice-Hall*, Inc., New Jersy, 1986.

[15] W.N. Lie. Automatic target segmentation by locally adaptive image thresholding, *IEEE Trans. on Image Processing*, vol. 41, no. 7, pp. 1036-1041, July 1995.

[16] Shamir, L. Human perception-based color segmentation using fuzzy logic, *International Conference on Image Processing, Computer Vision and Pattern Recognition* (IPCV 2006), vol. 2, pp. 496-505. Las Vegas, NV. 2006.

[17] Borji A. and Hamidi M. Evolving a fuzzy rule base for image segmentation, *Proceedings of world academy of science, engineering and technology*, vol. 22, July 2007, pp. 4-9.

[18] Estevez Pablo A., Folres Rodigo J., et al. Color image segmentation using fuzzy min-max neural networks, *Proceedings of international joint conference on neural networks*, August 2005, pp. 3052-3057.

[19] Karmakar, G., Dooley, L., Murshed, M. Fuzzy rule for image segmentation incorporating texture features, *Proceedings of international conference on image processing*, Vol.1, 2002, pp. 797-800.

[20] Chen Tie Qi and Lu Yi. Color image segmentation- an innovative approach. *Pattern recognition*, vol. 35, 2002, pp. 395-405.

[21] Tabakov Martin. A fuzzy clustering technique for medical image segmentation. *Proceedings of international symposium on evolving fuzzy systems*, September 2006, pp. 118-122.

[22] Ahmed M. N., Yamany S.M et al. A modified fuzzy c-means algorithm for bias field estimation and segmentation of MRI data, *IEEE transactions on medical imaging, Volume 21*, Issue 3, March 2002, pp. 193 – 199.

[23] Yang Y., Zheng Ch., and Lin P. Fuzzy c-means clustering algorithm with a novel penalty term for image segmentation. *Opto-electronic review*, Vol.13, Issue 4, 2005, pp. 309-315.

[24] Shen Shan, Sandham W., and Sterr A. MRI fuzzy segmentation of brain tissue using neighbourhood attraction with neural network optimization. *IEEE transactions on information technology in biomedicine*,Vol. 9, issue 3, September 2005, pp. 459-467.

[25] Kang Jiayin, Min Lequan et al. Novel modified fuzzy c-means algorithm with applications. *Digital Signal Processing*. March (2009),vol 19, no. 2, pp. 309-319.

[26] Ma Li and Stuanton R. C. A modified fuzzy c-means image segmentation algorithm for use with uneven illumination patterns, *Pattern recognition*, vol. 40, 2007, pp. 3005-3011.

[27] Tolias Yannis A. and Panas Stavros M. Image segmentation by a fuzzy clustering algorithm using adaptive spatially constrained functions, *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 28, Issue 3. May 1998, pp. 359-369.

[28] Mohamed N. A., Ahmed M. N., Modified fuzzy c-means in medical image segmentation, *Proceedings of IEEE international conference on Acoustics, Speech, and Signal Processing*, vol. 6, 1999, pp. 3429-3432.

[29] Kannan S.R. A new segmentation system for MR images based on fuzzy techniques, *Applied soft computing*, Vol. 8, Issue 4, September(2008), pp. 1599-1606.

[30] Yong Y., Chongxun Z., and Pan L. A novel fuzzy c-means algorithm for image thresholding, *Measurement Science review*, vol. 4, 2004, pp 11-19.

[31] Bezdek JC. , Cluster validity with fuzzy sets. *J Cybern*;3: 1974,58-73.

[32] Bezdek JC. Mathematical models for systematic and taxanomy. *Proceedings of Eighth International Conference on Numerical Taxonomy*, San Fransisco, 1975, pp. 143-166.

[33] Gonzalez R.C. and Woods R.E. Digital Image Processing *.2nd ed.Englewood Cliffs, NJ: Prentice –Hal*l; 2002.

[34] Pratt William K. Digital Image Processing, Third edition, *John Wiley and Sons*, 2001.

[35] Marr D. and Hildreth E.C. Theory of edge detection, *Proc. Of royal society of London.* B207,1980,187-217.

[36] Becerikli T., Anf Karan Y. A new fuzzy approach for edge detection. In J. Cabestany, A. Prieto, and D. Sandoval,editors, *Lecture Notes in Computer Science*, volume 3512, Springer-Verlag Berlin Heidelberg, 2005, pp. 943–951.

[37] Boskovitz V. and Guterman H. An Adaptive Neuro-Fuzzy System for Automatic image segmentation and edge detection. *IEEE Transactions on fuzzy systems*, vol. 10(2), April 2002, pp. 247–262.

[38] Hanmandlu M., See J., and Vasikarla S. Fuzzy edge detector using entropy optmization. In *Proceedings of the International Conference on Technology: Coding and Computing*, 2004, pages 665–670.

[39] Jhne B., HauBecker H., and GeiBler P. Handbook on Computer Vision and Applications, volume 2, chapter 22, Academic Press, 1999, pp. 684–728.

[40] Liang L., Basallo E., and Looney C. Image edge detection with fuzzy classifier. *In Proceedings of the ISCA 14th International Conference*, 2001, pages 279–283.

[41] Liang L. and Looney C. Competitive fuzzy edge detection. *Applied Soft Computing*, Vol. 3, 2003, pp. 132–137.

[42] Miosso C. J. and Bauchpiess A. Fuzzy inference system applied to edge detection in digital images. In *Proceedings of the V Brazilian Conference on Neural Networks*, 2001, pages 481–486.

[43] Hu L., Cheng H. D. and Zang M. A high performance edge detector based on fuzzy inference rules. *An International Journal on Information Sciences*, vol. 177, Nov 2007, no. 21, pp. 4768-4784.

[44] J. See, M. Hanmandlu, and S. Vasikarla. Fuzzy-based parameterized gaussian edge detector using global and local properties. In I. C. Society, editor, *Proceedings of the International Conference on Technology: Coding and Computing*, 2005, pages 101–106.

[45] Alshennawy Abdallah A. and Ayman A. Aly. Edge detection in digital images using fuzzy technique. *World Academy of Science, Engineering and Technology*. vol. 51, 2009, pp. 178-186.

[46] Md. Shoib Bhuiyan, Iwahori Y., and Iwata A. Optimal edge detection under difficult imaging conditions. *Technical report*, Educational Center for Information Processing and Dept. of Electrical and Computer Engineering, Nagoya Institute of Technology, Showa, Nagoya, 466-8555, JAPAN.

[47] Kwak S., Ko ByoungChul , et al. Salient human detection for robot vision. Pattern Analysis Application, vol. 10, March 2007, pp. 291-299.

[48] Tsai Piyu, Chang Chin-Chen, et al. An adaptive two-stage edge detection scheme for digital color images, Real-Time Imaging, vol. 8, no. 4, August 2002, pp. 329-343.

[49] Thompson Clay M. and Shure Loren. Image Processing Toolbox User's Guide - For Use With MatLab. *The MathWorks*, Inc., January 1995.

[50] Md. Shoib Bhuiyan, Matsuo H., Iwata A. et al. An improved neural network based edge detection method. *Technical report*, Educational Center for Information Processing and Dept. of Electrical and Computer Engineering, Nagoya Institute of Technology, Showa, Nagoya, 466-8555, JAPAN

[51] Boskovitz Victor and Guterman Hugo. An Adaptive Neuro –fuzzy System for Automatic Segmentation and Edge Detection. *IEEE Trans, On Fuzzy System*, vol. 10, no. 2, April(2002), pp. 247-261.

[52] Roger Jang J. S. and Gulley Ned. *Fuzzy Logic ToolboxUser's Guide*. The MathWorks, Inc., January 1995.

[53] Pal S. K. and King R. A. On edge detection of X-ray images using fuzzy sets, *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-5, no.1, pp.69-77, Jan.1983.

[54] Yi L. and Xue-quan C. An edge detection algorithm of remote sensing images based on fuzzy sets, *International conference on communications, circuits and systems,* ICCAS , vol. 2, 2004, pp. 984-988.

[55] Wang Q., Ruan H.B., and Chen H.F. Fast fuzzy edge detection algorithm, *Journal of image and graphics,* vol.6, no.1, Jan. 2001, pp. 92-95.

[56] Zhou D. L., Pan Q., and Zhang H. C. An improved algorithm of edge detection based on fuzzy sets, *Journal of image graphics*, vol. 6, no. 4, April 2001, pp. 353-358.

[57] Canny J. F., A computational approach to edge detection, *IEEE Trans. Pattern Analysis Mach. Intell.,* vol.PAMI-8, no. 6, Nov. 1986, pp. 679-698.

[58] C.I.E Colorimetry Committee Proposal for study of color spaces, Tech. Note, *J. Opt. Soc. Am*, 64, 6,June 1974,896-897.

[59] Tolias Y.A. and Panas S.M. On applying spatial constraints in fuzzy image clustering using a fuzzy rule based system, *IEEE signal processing letters*, 5(1998):pp. 245-247.

[60] Noordam J.C., van den Broek W.H.A.M., and Buydens L.M.C. Geometrically guided fuzzy c-means clustering for multivariate image segmentation. *Proc. Int. Conf. on Pattern Recognition* 1, (2000),pp. 462- 465.

[61] Liew A.W.C., Leung S.H., and Lau W.H. Fuzzy image clustering incorporating spatial continuity, *IEEE Proc. Visual Image Signal Process*. 147, 185.192 (2000).

[62] Siyal M. Y. and Yu Lin. An intelligent modified fuzzy c-means based algorithm for bias estimation and segmentation of brain MRI. Pattern Recognition Letters. vol. 26, no. 13, Oct. 2005, pp. 2052-2062.

[63] Li X., Li L., Lu H., Chen D., and Liang Z. Inhomogeneity correction for magnetic resonance images with fuzzy c-mean algorithm., *Proc. SPIE 5032*, 995.1005 (2003).

[64] Kwon M.J., Han Y.J., Shin I.H., and Park H.W. Hierarchical fuzzy segmentation of brain MR images., *Int. J. Imaging Systems and Technology* ,vol. 13, (2003),pp.115-125.

[65] Pham D.L. and Prince J.L., .Adaptive fuzzy segmentation of magnetic resonance images. *IEEE Trans. Medical Imaging*, vol. 18, (1999), pp.737-752.

[66] Bezdeck J.C. Pattern recognition with fuzzy objective function algorithms. *New York, plenum Press*, 1981.

[67] Sherrier R.H. and Johnson G.A. Regionally adaptive histogram equalization of the chest. *IEEE Trans. Med. Image*, Mi- 6 (1987), pp. 1-7.

[68] Pizer S.M., Amburn E.P., et al., Adaptive Histogram equalization and its variations, *Computer Vision , Graph Image Process*. 39, (1987), pp. 355-368.

[69] L. Pham Dzung. Spatial models for fuzzy clustering, *Computer Vision and Image Understanding*, vol. 84**,** 2001, pp. 285–297.

[70] Acton S. T. and. Mukherjee D. P. Scale space classification using area morphology, *IEEE Trans. Image Process*., vol. 9. no. 9, pp. 623-635, Apr.2000.

[71] Dave R. N. R. N. Characterization and detection of noise in clustering, *Pattern Recognit. Lett.*, vol.12, pp. 657-664, 1991.

[72] Krishnapuram R. R. and Keller J. M. A possibilistic approach to clustering, *IEEE Trans. Fuzzy Syst.*, vol. 1, no. 2, pp.98-110, May 1993.

[73] Pedrycz W, Waletzky J. Fuzzy Clustering with partial supervision. *IEEE Trans. Syst Man Cybern Part B Cybern* 1997; 27:787-95.

[74] Chuang Keh-Shih, Tzeng Hong-Long, Chen Sharon et al., Fuzzy C-means clustering with spatial information for image segmentation, *Computeried Medical Imaging and graphics*, 30(2006) 9-15.

[75] Yang Zhang, Chung Fu-Lai, et al, Robust fuzzy clustering-based image segmentation, *Applied Soft Computing*, vol(9), no. 1, Jan (2009), pp. 80-84.

[76] Karmakar G.C, Dooley L. and Rahman S.M. A survey of fuzzy rule based image segmentation techniques, *1st IEEE Pacific-Rim Conf. on Multimedia, Sydney, Australia* (2000), pp. 350–353.

Contributions by Candidate

1. **K. Nirulata**, S.Meher, "An Efficient Edge Detector for Digital Images"**,** *Proceedings of national conference on devices, intelligent systems and communications, MITDISC-2007,MIT-Manipal,7th-8th dec-2007.*

2. **K. Nirulata**, S.Meher, "Color Image segmentation using Fuzzy Clustering"**,** *Proceedings of International Conference on Emerging Technologies and Applications in Engg., Tech. and Sciences ,ICETAETS-2008,Saurastra University,Rajkot,13th-14th Jan-2008.pg.no.-1105-1109.*

3. **K. Nirulata**, S.Meher, "Segmentation Of Unevenly Illuminated Color Images"**,** *Proceedings of IEEE sponsored conference on Computational Intelligence, Control and Computer vision in Robotics and Automation, CICCRA-2008,NIT-Rourkela,pg.no.23-27.*

4. **K. Nirulata**, S.Meher, "Segmentation of Noisy Color Images"**,** *Accepted for publication in International Journal of Applied Artificial Intelligence in Engineering System.*

5. **K. Nirulata**, S.Meher, "Segmentation of Noisy Color Images using Neighborhood property of Digital Image"**,** *Accepted for publication in International Journal of Computer Science and Management System*.

6. **K. Nirulata**, S.Meher, "Skin Tumor Segmentation using Fuzzy c-means Clustering with Neighbourhood Attraction"**,** *Communicated to International Journal of Computers and Electrical Engineering.*